VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

ROZHRANÍ BEZPEČNOSTNÍHO SYSTÉMU PRO WIFI SÍTĚ

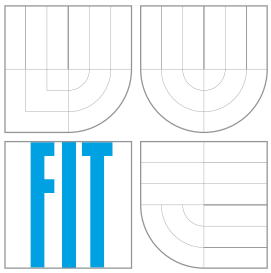
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. PETR HIRŠ

BRNO 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

ROZHRANÍ BEZPEČNOSTNÍHO SYSTÉMU PRO WIFI SÍTĚ

USER INTERFACE FOR WIFI NETWORK SECURITY SYSTEM

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — Bc. PETR HIRŠ
AUTHOR

VEDOUCÍ PRÁCE — doc. Ing. DANIEL CVRČEK, Ph.D.
SUPERVISOR

BRNO 2007

Zadání diplomové práce/5087/2006/xhirsp00

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav inteligentních systémů Akademický rok 2006/2007

Zadání diplomové práce

Řešitel: **Hirš Petr, Bc.**
Obor: Inteligentní systémy
Téma: **Rozhraní bezpečnostního systému pro WiFi sítě**
Kategorie: Bezpečnost

Pokyny:

1. Seznamte se s existujícími diplomovými pracemi ohledně reputačních systémů pro WiFi sítě, s technologiemi pro návrh webových rozhraní a s databázovými systémy (mySQL, Oracle, PostgreSQL, příp. dalšími).
2. Proveďte analýzu dat, které jsou získávány o provozu WiFi sítí a jejich vnitřních závislostí.
3. Navrhněte databázové schéma a rozhraní pro komunikaci s uživateli. Rovněž je třeba navrhnout vnitřní rozhraní pro komunikaci se systémem sbírajícím data.
4. Implementujte databázovou podporu pro zpracování dat získaných o provozu WiFi sítě.
5. Vytvořte nové grafické uživatelské rozhraní - včetně kreslení topologie sítě, možnosti nastavování firewallů na jednotlivých přístupových bodech (AP). Proveďte alespoň přípravu rozhraní pro možné nastavování QoS.
6. Proveďte srovnání vytvořeného systému s jinými existujícími aplikacemi, shodnoťte dosažené výsledky.

Literatura:

- Podel specifikace vedoucího.

Při obhajobě semestrální části diplomového projektu je požadováno:

- Body 1 a 2 a rozpracování bodů 3-5.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Cvrček Daniel, doc. Ing., Ph.D.**, UITS FIT VUT
Datum zadání: 28. února 2006
Datum odevzdání: 22. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav inteligentních systémů
612 66 Brno, Božetěchova 2

L.S.

doc. Dr. Ing. Petr Hanáček
vedoucí ústavu

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Bc. Petr Hirš**
Id studenta: 49155
Bytem: Přísecká 115, 257 68 Dolní Kralovice
Narozen: 03. 01. 1981, Ledeč nad Sázavou
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
diplomová práce

Název VŠKP: Rozhraní bezpečnostního systému pro WiFi sítě
Vedoucí/školitel VŠKP: Cvrček Daniel, doc. Ing., Ph.D.
Ústav: Ústav inteligentních systémů
Datum obhajoby VŠKP: .....

VŠKP odevzdal autor nabyvateli v:
tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
   - ☐ ihned po uzavření této smlouvy
   - ☐ 1 rok po uzavření této smlouvy
   - ☐ 3 roky po uzavření této smlouvy
   - ☐ 5 let po uzavření této smlouvy
   - ☐ 10 let po uzavření této smlouvy

   (z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: .....

.....	.....
Nabyvatel	Autor

Abstrakt

V diplomovém projektu jsem se zaměřil na analýzu současné implementace reputačního systému. Zabývám se především analýzou a návrhem možných rozšíření uživatelského rozhraní reputačního systému. V analýze zmiňuji alternativy řešení a porovnávám své řešení s podobnými systémy pro zabezpečení sítí. Část semestrálního projektu je věnovaná návrhu nového systému uživatelského rozhraní. Zmiňuji návrh databáze a způsob přihlašovaní uživatelů. Několik kapitol je věnováno řešení odstínění obsahu a prezentace dat, které vedou k použití šablonovacího systému. Projekt dále dokumentuje zajímavé kroky, které se objevily v průběhu implementace. V závěru projektu uvádím celkové zhodnocení projektu a několik zmínek o možných rozšířeních.

Klíčová slova

wifi sítě, reputační systém, uživatelské rozhraní, přihlašování uživatelů, šablonovací systém, jazykové mutace, topologie sítě, grafy

Abstract

This master thesis deals with analysis of an implementation of a reputation system. I particularly focus on analysis of user interface and to find an approach for possible user interface improvements of such a reputation system. I describe possible solutions and I also compare my solution with similar existing systems that are also focused on network security. A part of the diploma thesis is devoted to user interface design. There is also mentioned a database design together with a way the users log on into the system. A few chapters are devoted to the area of data presentation and the usage of template system. In the work there are also described interesting issues that appeared during the implementation. In the conclusion of this work, there is an overall project evaluation and also several remarks about the possible upgrade.

Keywords

wifi networks, reputation system, user interface, user login, template system, language mutation, net topology, charts

Citace

Petr Hirš: Rozhraní bezpečnostního systému pro WiFi sítě, diplomová práce, Brno, FIT VUT v Brně, 2007

Rozhraní bezpečnostního systému pro WiFi sítě

Prohlášení

Prohlašuji, že jsem tento diplomový projekt vypracoval samostatně pod vedením doc. Ing. Daniela Cvrčka, Ph.D. Další informace mi poskytl Ing. Petr Blahák. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Petr Hirš

22. 5. 2007

Poděkování

Děkuji doc. Ing. Danielu Cvrčkovi, Ph.D. za odborné vedení, cenné rady a pomoc při zpracování diplomového projektu. Dále bych chtěl poděkovat Bc. Petru Kaněčkovi za pomoc při sazbě tohoto dokumentu.

Obsah

Kapitola 1

Úvod

Mobilní komunikace je jedním ze společných jmenovatelů současné doby. Lidé se nechtějí vázat a rádi využívají možnosti připojit se na Internet a přečíst si novou poštu, aniž by museli vyhledat klasický počítač připojený do internetu. Tyto a nespočet dalších požadavků dnešní doby umožnily velice rychlý a masivní nástup technologie zvané *WiFi*[1].

1.1 WiFi sítě

V oblasti bezdrátových sítí *WLAN*[2] se lze setkat se spoustou norem a zkratek, které se k této problematice pojí. Některé pojmy se natolik používají, že je lidé začínají používat i v jiném smyslu, než který doopravdy vyjadřují. Pokusím se teď v několika větách shrnout ty nejpoužívanější a uvést jejich správný význam.

V prvopočátku byla bezdrátová komunikace věcí proprietálních řešení různých výrobců a v drtivé většině jednotlivá bezdrátová zařízení komunikovala pouze se zařízeními stejného výrobce. Tato situace se zlepšuje až přijetím normy IEEE 802.11.

V dnešní době jsou nejrozšířenější sítě, které respektují normy IEEE 802.11b a IEEE 802.11g pro nelicencované pásmo 2,4 GHz. Vzhledem k velkému rozšíření těchto sítí a sním spojené zarušení přecházejí nově budované spoje do pásma 5 GHz. Komunikaci v pásmu 5 Ghz určuje norma IEEE 802.11a.

Ono zmiňované označení WiFi se používá pro certifikaci zařízení pracujících dle standardů IEEE 802.11a/b/g. Zařízení, která nesou označení WiFi, by měla mezi sebou bezproblémově pracovat i v případě, když jednotlivá zařízení vyrábí různí výrobci.

WiFi sítě a jejich bezpečnost je velice často diskutovaný problém. Vzhledem k tomu, že se data v těchto sítích vysílají všesměrově, není problém je odposlechnout. Na řadu tedy přichází šifrování, které je v prvních standardech řešeno pomocí protokolu *WEP*[3]. Bohužel použití slabého šifrovacího klíče umožňuje jeho brzké prolomení. Novější standardy

[1] z anglického Wireless Fidelity
[2] z anglického Wireless Local Area Network
[3] z anglického Wired Equivalent Privacy

pamatují na tuto slabinu a řeší ji pomocí novějších zabezpečovacích mechanismů *WPA*[4] a dalších rozšíření.

Nepříjemná situace se zabezpečením WiFi sítí ale stále přetrvává. Je to způsobeno masovým rozšířením hardwaru, který buď podporuje pouze staré způsoby zabezpečení, nebo existují případy, kdy je třeba zachovat přístup do sítě i pro klienty se zastaralými přístupovými kartami. Dalším problémem může být prozrazení tajného klíče klientem sítě někomu, kdo by za normálních okolností do sítě přístup neměl. Toto velice znepříjemňuje provoz WiFi sítí a není jednoduché odlišit, který uživatel sítě, je oprávněný využívat služeb sítě, a který ne. Tento problém se snaží řešit tzv. reputační systémy.

1.2 Obecný princip funkčnosti reputačních systémů

Reputační systém je pojem, který v sobě mísí pojmy sociologie a pojmy ryze technické. Funkčnost systému je postavena na vzájemném hodnocení klientů, nebo chcete-li účastníků systému, mezi sebou. Výsledné ohodnocení klienta je označováno jako reputační kredit. Tento kredit ostatním klientům říká, do jaké míry je možné tomuto klientovi důvěřovat, i když tohoto klienta přímo neznají. Reputační kredit je dynamická vlastnost, která se mění dle toho, jak se daný klient v systému chová.

Veškeré reputační systémy berou jako základ pro hodnocení klienta jeho každodenní chování v dané komunitě. V následujícím textu se budu snažit vymezit několik základních pravidel, které platí mezi lidmi v tzv. sociální síti. Tyto pravidla jsou základním stavebním kamenem již zmiňovaných reputačních systémů.

Lidé při řízení vztahů mezi sebou používají úplně jiné metody, než používají počítače pro řízení bezpečnosti. Počítače komunikaci mezi sebou určují pomocí přesné identifikace druhého bodu komunikace a nalezení nastavených práv v příslušné databázi. Kdežto lidé si v průběhu svého života (celého, nebo jen části života v určité komunitě) vytvářejí tzv. sociální síť.

Součástí této sociální sítě jsou lidé, ke kterým si člověk vytvořil nějaký vztah založený na předchozí zkušenosti s nimi. Mohou to být přátelé, známí, ale i nepřátelé apod. V takto vybudované sociální síti existuje několik úrovní důvěry. Jinak si člověk hodnotí lidi, které viděl jednou v životě, a na jiné úrovni přistupuje k lidem, které označuje jako své přátele. Úrovně v této sociální síti si lze představit jako jisté stupně důvěryhodnosti daných subjektů. Na základě této vybudované úrovně důvěryhodnosti se pak člověk rozhoduje, zda vyhoví, nebo nevyhoví požadavkům jiného člověka.

Jak je uvedeno v [1], získávání informací a určení důvěryhodnosti v reputačním systému lze rozdělit mezi několik následujících částí.

[4] z anglického WiFi Protected Access

1.2.1 Využití informace založené na předchozí zkušenosti se subjektem

Jedná se o hlavní zdroj informací pro obecný reputační systém. Důležité je, aby se informace o důvěrnosti měnily dynamicky. Tímto lze poté pokrýt případy, kdy se důvěryhodný účastník systému rozhodne chovat v rozporu s pravidly systému. Nebo také naopak, kdy si nedůvěryhodný účastník díky svému počínaní vylepšuje reputaci.

1.2.2 Rozhodnutí se na základě výše způsobené škody

Jedná se o případy, kdy systém podstoupí jisté riziko tím, že danému účastníkovi vyhoví v jeho požadavku. Nebo se může jednat o případ, kdy systém rozhodne, že účastníkovi nevyhoví, protože by případné škody byly větší, než je možné v současné době akceptovat. V běžném životě je takovéto chování možné pozorovat např. při žádosti o vyřízení půjčky.

1.2.3 Informace od jiných důvěryhodných subjektů

V praxi toto znamená, že pokud systém musí komunikovat s někým, o kterém nemá žádné předchozí informace, může požádat třetí stranu o informaci, jak je daný účastník důvěryhodný. Třetí strany jsou v reputačním systému také hodnoceny tím, do jaké míry je možné informacím, které tyto třetí strany poskytují.

1.2.4 Informace od neznámých subjektů

Jedná se o poslední a zároveň asi nejhorší možnost, jak získat informace o účastníkovi systému. Objektivnost takto získané informace roste s počtem odpovědí dotázaných subjektů. Je ale třeba počítat s variantou, kdy dotázané subjekty záměrně odpovídají nesprávně a ve prospěch účastníka.

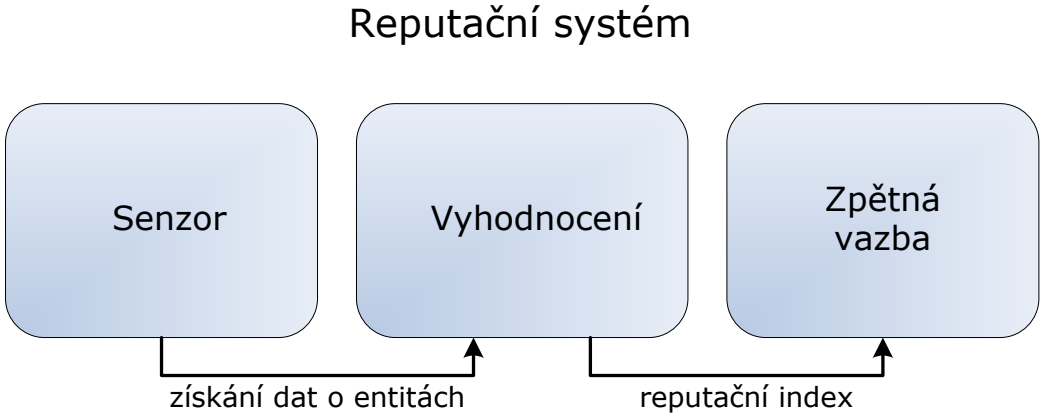
1.3 Reputace a reputační systémy ve WiFi sítích

Výpočet reputačního kreditu, neboli ohodnocení chování uživatelů sítě, může vést ke klasifikaci uživatelů dle různých hledisek např. na důvěryhodné, nedůvěryhodné, právoplatné, nebo neprávoplatné uživatele sítě. Takováto klasifikace uživatelů může být velice užitečná hlavně provozovatelům sítí, kteří se snaží tyto sítě zabezpečit a poskytnout všem právoplatným klientům své služby. Naopak neoprávněné uživatele je třeba identifikovat a zamezit jejich přístup ke službám provozované sítě.

Reputační systémy mají právě za úkol sledovat údaje o uživatelích a uchovávat jejich reputační kredit. Tento kredit se v čase mění a slouží pro ostatní uživatele systému jako jediná charakteristika, která určuje jejich důvěryhodnost v systému.

Obecný reputační systém uvedený na obrázku číslo 1.1 lze rozdělit do třech základních částí. Jedná se o senzorovou, hodnotící a zpětnovazebnou část. Senzorová část má za úkol získávat data o chování určité entity. Hodnotící část shromažďuje informace ze senzorů

a vypočítává již zmiňovaný reputační kredit. Zpětnovazební část umožňuje reputačnímu systému reagovat na základě vypočteného reputačního kreditu.



Obrázek 1.1: Schéma reputačního systému

1.3.1 IDS, IPS a reputační systém

Pro řešení bezpečnosti sítí se používají tzv. IDS[5] a IPS[6] systémy. V následujícím textu se pokusím vysvětlit tyto pojmy a určit, kam reputační systém řešící bezpečnost dle tohoto dělení patří. Informace o IDS a IPS systémech jsem čerpal z opor předmětu *„Návrh, správa a bezpečnost“* vyučovaného na FIT VUT v Brně.

Systémy IDS provádí detekci síťových útoků. Využívají monitorování a analýzy aktivity uživatelů. Dokáží rozpoznat známé útoky, provádět audity systémové konfigurace a statisticky analyzovat abnormální aktivity v síti.

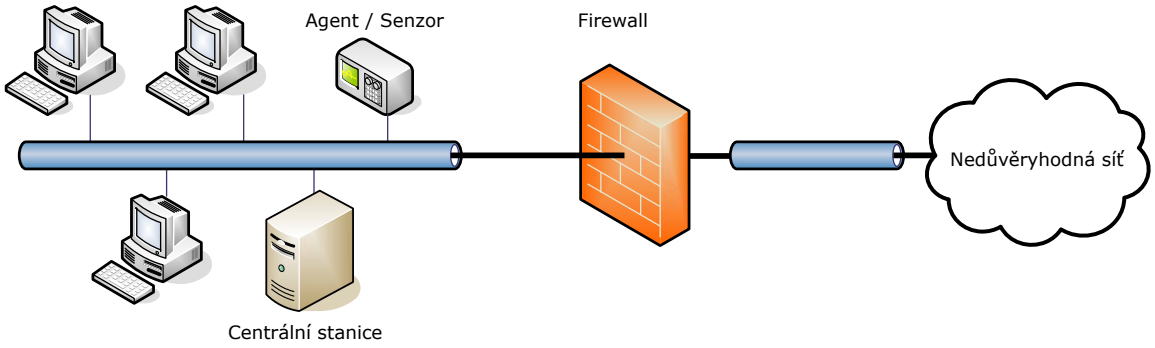
IPS systémy se snaží prevencí zastavit nebo minimalizovat útok na síťové zdroje. Takovéto systémy detekují útok na síťové zdroje. Při jeho výskytu zastaví útok a přenastaví současný systém tak, aby byl v budoucnu proti takovému útoku odolnější.

Oba systémy IDS i IPS se dále mohou dělit na tzv. systémy *Host-based* a *Network-based*. U systému typu Host-based se detekce týká pouze jednotlivých počítačů. Na každém počítači musí být přítomen tzv. agent, který sbírá informace o stavu systému a posílá stavové informace centrální stanici.

U architektury systému typu Network-based se používá pro zjištění průniku aktivní způsob. Tím je myšleno odchytávání a analýza síťového provozu. Toto zajišťují tzv. senzory, které jsou umístěny do sledované sítě. Senzor může být jak specializované zařízení, tak i specializovaný program.

Pro úplnost uvádím na obrázku 1.2 obecnou architekturu IDS a IPS systémů.

[5] z anglického Intrusion Detection System
[6] z anglického Intrusion Prevention System



Obrázek 1.2: Obecná architektura IDS a IPS systémů

Pokud vezmeme v potaz zmiňované dělení bezpečnostních systémů, reputační systém se pohybuje někde mezi systémem IDS a IPS. Reputační systém se nesnaží přímo detekovat útoky dle známých vzorů ani neprovádí audity. Naopak po vzoru IDS a IPS obsahuje senzor, který sbírá data o dění v síti a vytváří statistiky, dle kterých poté jádro reputačního systému určuje reputační kredit.

IDS systémy je velice vhodné nasadit pro zlepšení bezpečnosti sítě. IDS dokáží úspěšně blokovat řadu DoS[7] útoků, útoky pomocí záplav SYN paketů a další. Druhou stránkou problému nasazení IDS systému může být pořizovací cena. Jak uvádí Petr Blahák v [2], zejména při nasazení IDS v bezdrátových sítích, je nutné ke každému přístupovému bodu pořídit další anténu a bezdrátovou kartu, která bude daný prostor zajišťovat. Reputační systém se nesnaží zastoupit funkci IDS systémů, ale měl by být především doplňkem těchto systémů.

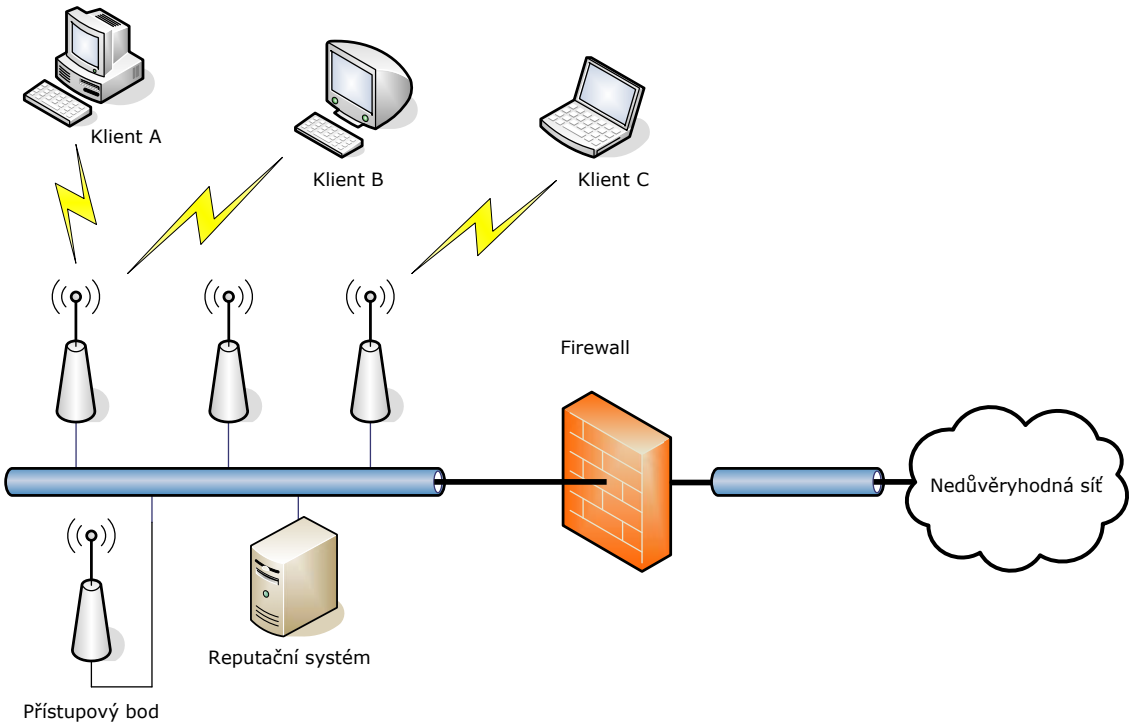
Myšlenka využití reputačního systému pro hodnocení uživatelů WiFi sítí pochází od Petra Blaháka, na jehož diplomovou práci [2] tento diplomový projekt navazuje. Více o tom, jak reputační systém funguje, uvádím v kapitole zabývající se analýzou současného systému. Detaily týkající se implementace a testování a nasazení systému jsou uvedeny právě v [2] a v této práci nebudou zmiňovány.

Uvedu pouze, že pro funkční reputační systém je nutné zabezpečit **sběr dat**, které se v čase mění a charakterizují provoz v síti popř. přímo popisují jednotlivé klienty. Tyto informace lze ve WiFi sítích získat z přístupových bodů, které v dnešní době mají v drtivé většině operační systém, nejčastěji speciálně upravenou distribuci operačního systému linux.

Vyhodnocení takto získaných dat provádí počítač, který je k tomuto účelu do sítě umístěn. Zpravidla na tomto počítači bývá umístěno jádro reputačního systému a úložiště pro získaná data.

[7] z anglického Denial of Service, jedná se o útok, který vede k vyčerpání systémových nebo síťových zdrojů postiženého systému

Pro řešení **zpětné vazby** reputačního systému se používá rekonfigurace přístupových bodů reputačního systému. Může se jednat o různá nastavení QoS[8] nebo např. změny v nastavení firewallu. Obecné schéma sítě s reputačním systémem tak, jak je implementován Petrem Blahákem, je uvedeno na obrázku 1.3.



Obrázek 1.3: Obecné schéma sítě s reputačním systémem

1.4 Vymezení cíle diplomového projektu

Zmiňovaný reputační systém aplikovaný na reputaci WiFi sítí a jeho implementace jsou popsány v diplomové práci [2]. V mé diplomové práci se budu snažit vylepšit a rozšířit stávající implementaci uživatelského rozhraní reputačního systému.

Mezi hlavní cíle diplomové práce patří seznámení se současnou implementací reputačního systému. Následuje analýza dat, které jsou získávány ze senzorů reputačního systému. Na základě této analýzy bude současný systém rozšířen o robustnější databázový systém s kompletně novým návrhem databázového schématu. Volba nové databáze by měla umožnit nasazení systému pro rozsáhlejší sítě a současně s převodem výpočtů do databázové vrstvy zrychlit celý reputační systém.

[8] kvalita služby z anglického Quality of Service vyjadřuje jeden z trendů vývoje technologií a služeb počítačových sítí - poskytovat uživatelům služby s definovanou kvalitou

Přepracování datového úložiště také souvisí se změnou současných skriptů pro získávání dat z jednotlivých přístupových bodů. Současné řešení projektu bude komunikovat se zařízeními založenými na systému RouterOS.

Hlavním cílem této práce je vytvořit nové, uživatelsky přívětivé, grafické rozhraní, které zastřeší veškeré funkce reputačního systému. Budoucí uživatelské rozhraní by mělo přehledně zobrazit informace o všech přístupových bodech a jejich klientech. Umožní ze získaných dat o přístupových bodech automatické generování topologie sítě. Samozřejmostí bude zobrazení reputací jednotlivých klientů a doporučení restriktivních kroků u klientů označených jako neprávoplatní uživatelé sítě. Součástí uživatelského rozhraní budou i prvky pro nastavení jednotlivých služeb přístupových bodů a prvky tvorbu grafů z uložených dat systému.

Uživatelské rozhraní by mělo být dostupné pomocí sítě Internet, aby bylo možné jednoduchým způsobem vzdáleně dohlížet na síť reputačního systému. Uživatelské rozhraní musí umožňovat přihlašování uživatelů a nastavení jejich oprávnění k operacím reputačního systému. Systém uživatelského rozhraní by měl být vhodně navržen pro další případná rozšíření a měl by podporovat jazykové mutace.

1.5 Stručný obsah práce

V diplomové práci jsem se zaměřil na analýzu současné implementace reputačního systému. Zabývám se především analýzou a návrhem možných rozšíření uživatelského rozhraní reputačního systému. V analýze zmiňuji alternativy řešení a porovnávám své řešení s podobnými systémy pro zabezpečení sítí.

Část diplomového projektu je věnovaná návrhu nového systému uživatelského rozhraní. Zmiňuji návrh databáze a způsob přihlašovaní uživatelů. Několik kapitol je věnováno řešení odstínění obsahu a prezentace dat, které vedou k použití šablonovacího systému.

Projekt dále dokumentuje zajímavé kroky, které se objevily v průběhu implementace, včetně popisu způsobu implementace grafů, které systém používá pro vizualizaci dat.

V závěru projektu uvádím celkové zhodnocení projektu a několik zmínek o možných rozšířeních.

Kapitola 2

Analýza

V současné době neexistují reputační systémy, které se zabývají reputací WiFi sítí pro řešení bezpečnosti těchto sítí. Nabízí se srovnávat toto řešení se systémy pro detekci průniků. Tyto systémy mají některé společné rysy, které budu v analýze porovnávat s požadavky na vyvíjený systém. Pro tuto analýzu jsem vybral produkty, ze kterých vycházela i původní implementace reputačního systému.

Jedná se o produkty Airdefense RougeWatch a Airdefense Guard [3], programy SnortWireless [4] a WIDZ [5]. Samozřejmostí je analýza současného systému [2], na který tento diplomový projekt navazuje. V projektu tedy budou použité některé postupy, algoritmy popř. skripty původního reputačního systému. Při formování požadavků na rozšíření současného systému, bude brán v potaz i v současné době nový monitorovací program Dude [6] firmy MikroTik.

V následujících podkapitolách uvádím obecné závěry vyplývající z analýzy zmiňovaných systémů. Snažím se vymezit společné rysy analyzovaných systémů a současného systému pro reputaci, které lze využít při návrhu a implementaci rozšíření pro současný systém.

2.1 Analýza současné implementace reputačního systému

Projekt je implementován ve třech základních částech. Jedná se o části *získání dat z přístupových bodů*, *vyhodnocení získaných dat* a implementace *zpětné vazby* systému. Jednotlivé bloky systému, pokud to tak můžeme nazvat, spolu komunikují pomocí soborů ve formátu ARFF[1].

Soubor ARFF představuje standardní reprezentaci souboru dat, která se skládá z nezávislých, nijak neseřazených instancí a neobsahuje vzájemný vztah mezi instancemi. ARFF soubor byl vyvinut v rámci projektu strojového učení na oddělení Computer Science na universitě University of Waikato pro použití v softwaru Weka[2].

[1] z anglického Attribute-Relation File Format

[2] jedná se o open source projekt naprogramovaný v jazyku Java, který se věnuje dolování dat

Jádro reputačního systému tvoří skripty napsané v shellu s využitím dalších příkazů jako jsou sed, awk, grep a další. Tyto skripty obstarávají získávání dat z přístupových bodů. Sběr dat z přístupových bodů je automatizován. Skript *data_apX.sh* je volán automaticky každých 15 minut pomocí programu Cron[3].

2.1.1 Vyhodnocení získaných dat

Vyhodnocení získaných dat z přístupových bodů provádí systém z několika pohledů. Při implementaci systému bylo zvažováno hodnocení klientů dle *doby připojení klienta*, *množství přenesených dat*, *odchylky síly signálu* a dle *ověření dat proti databázi* s daty o klientovi z doby jeho registrace do sítě.

Současný stav reputačního systému používá k hodnocení klientů pouze 2 metody. První metodou je ověření získaných dat proti databázi. Tato metoda spočívá v existenci databáze klientů dané komunitní sítě. Při registraci nového klienta do sítě je klientovi přiřazena MAC a IP adresa. Pokud je z vyčtených informací přístupového bodu zjištěna neshoda těchto údajů, je daný klient pomocí modulu zpětné vazby ze sítě odpojen.

Druhou metodou využitou v reputačním systému pro hodnocení klientů je hodnocení klienta dle odchylky síly signálu. Testováním bylo zjištěno, že průměrná hodnota a směrodatná odchylka síly signálu pro daného klienta se v dlouhodobém intervalu téměř nemění. Díky této vlastnosti reputační systém doporučuje odpojit klienty, kterým se tato hodnota skokově mění. Systém úmyslně klienty neodpojuje od sítě automaticky, protože se může stát, že kolísání síly signálu způsobilo nějaké zarušení nebo připojení mobilního zařízení, které mění svou polohu.

Doba připojení klienta do sítě a množství přenesených dat klientem nejsou použity jako kritéria pro výpočet reputace klientů sítě. Tyto kritéria při testování vedla k příliš nepřesným výsledkům a doporučením k odpojení pravoplatných klientů. Více o tomto problému uvádí Petr Blahák ve své práci [1].

Hodnocení reputace klientů jen podle dvou zmiňovaných hledisek se jeví jako nedostatečné. Kvalitnější a přesnější výpočet reputace je tedy předmětem dalšího vývoje systému a svým tématem přesahuje rozsah a zaměření této práce.

2.1.2 Zpětná vazba systému

Pro řešení zpětné vazby reputačního systému Petra Blaháka bylo použito úplné odpojení klienta od sítě. Žádost o odpojení klienta je opět generována do souboru ARFF. Tento soubor je přenesen na příslušný přístupový bod, kde je skriptem zpracován. Odpojení klienta od sítě se provede pomocí příkazu *iptables*.

[3]CRON je systémový nástroj pro spouštění různých programů v předem definovaném čase a intervalech

2.1.3 Analýza současného uživatelského rozhraní aplikace

Pro implementaci uživatelského rozhraní zvolil autor systému webové stránky doplněné skripty v jazyku PHP. Implementované uživatelské rozhraní je nedostačující a troufám si říci, že nesplňuje téměř žádná pravidla o použitelnosti, více na [8], webových stránek. Uživatelské rozhraní nemá pevně daný řád a prvky uživatelského rozhraní jsou evidentně umístěny bez dlouhého přemýšlení. Co se týká barev a grafického návrhu, také téměř žádný neexistuje.

Základní grafy autor projektu řešil pomocí programu PHP/SWF charts. Jedná se o pěkně vypadající řešení, bohužel není možné grafy jakýmkoliv způsobem více měnit a je nutné se spolehnout na implementaci, kterou dodal výrobce.

Uživatelské rozhraní vůbec neumožňuje přihlašování uživatelů a řízení jejich práv. Není ani možné využít podporu pro jazykové mutace. Implementace uživatelského rozhraní nevyužívá žádnou možnost oddělení formy od obsahu. Kód internetových stránek je psán nečistě a kombinuje přímé a externí kaskádové styly. U skriptů PHP, které obstarávají data pro vykreslení stránek, chybí komentáře a jakákoliv štábní kultura.

Stav uživatelského rozhraní projektu přisuzuji tomu, že autor se ve své práci zabýval především návrhem a implementací reputačního systému. V uživatelském rozhraní bylo pravděpodobně vytvořeno pouze jen to nejnutnější, aby bylo možné prezentovat výsledky reputačního systému. Pro úplnost uvádím v příloze A ukázku původního uživatelského rozhraní.

2.2 Analýza nároků reputačního systému

Reputační systém musí dohlížet na provoz sítě nepřetržitě 24 hodin denně, aby bylo možné pracovat s aktuálními informacemi o celé síti. Uživatelské rozhraní řídící aplikace musí být přístupné kdekoliv ze sítě Internet. Je třeba zajistit funkčnost rozhraní při práci i několika uživatelů současně. Z tohoto požadavku nepřímo vyplývá zajištění autentizace uživatelů a řízení jejich práv v systému uživatelského rozhraní.

Systém pro reputaci klade vysoké nároky na datové úložiště, ve kterém shromažďuje informace o stavu sítě. Jedná se jak o aktuální informace o stavu sítě, tak o informace s historickou vypovídající hodnotou. Je třeba zvolit takové úložiště, které zajistí bezpečnost dat, protože se může jednat o citlivá data vztahující se k jednotlivým uživatelům sítě. Zřetel je také třeba brát na rychlost datového úložiště. Nad uloženými daty budou převažovat výběrové operace se sumarizačními dotazy a dotazy, které spojují několik různých druhů dat. Tyto dotazy bude používat uživatelské rozhraní pro vytvoření statistik, kreslení grafů apod. Struktura dat v úložišti musí být volena vhodně, aby skripty, které získávají data z jednotlivých přístupových bodů sítě, pracovaly rychle a nezatěžovaly celý reputační systém.

Z pohledu uživatele reputačního systému by měl být systém nenáročný na obsluhu a přehledně zpracovaný. Pro přístup k systému musí postačovat připojení k Internetu a základní programové vybavení pro práci s Internetem.

2.3 Analýza požadavků na uživatelské rozhraní

Uživatelské rozhraní současné implementace reputačního systému je velice strohé a umožňuje pouze zobrazení informací o reputaci. Bez dobrého uživatelského rozhraní, které umožní uživateli získat přehled o celé síti a řídit nastavení jednotlivých přístupových bodů, nemůže ani výborný systém fungovat.

Jako vzor pro řešení uživatelského rozhraní lze využít již zmiňované produkty firmy AirDefense [3] nebo aplikaci Dude [6] firmy MikroTik. Aplikace pro ucelený přehled o řízené síti zobrazují nákres topologie sítě s různou hloubkou zobrazených detailů. Pro zobrazení statistických dat jsou použity různé typy grafů, které přehledně reprezentují sledované vlastnosti systému.

Nové uživatelské rozhraní reputačního systému musí přehledně podávat informaci o celé síti reputačního systému. Součástí uživatelského rozhraní by měla být nápověda nebo doporučení pro práci se systémem. Uživatelské rozhraní by mělo být možné jednoduše rozšiřovat o další ovládací prvky, a je tedy třeba dobře zvolit architekturu celé aplikace. Je třeba, aby uživatelské rozhraní podporovalo jazykové mutace, které je možné jednoduše rozšířit o další jazyk.

2.3.1 Tvorba grafů

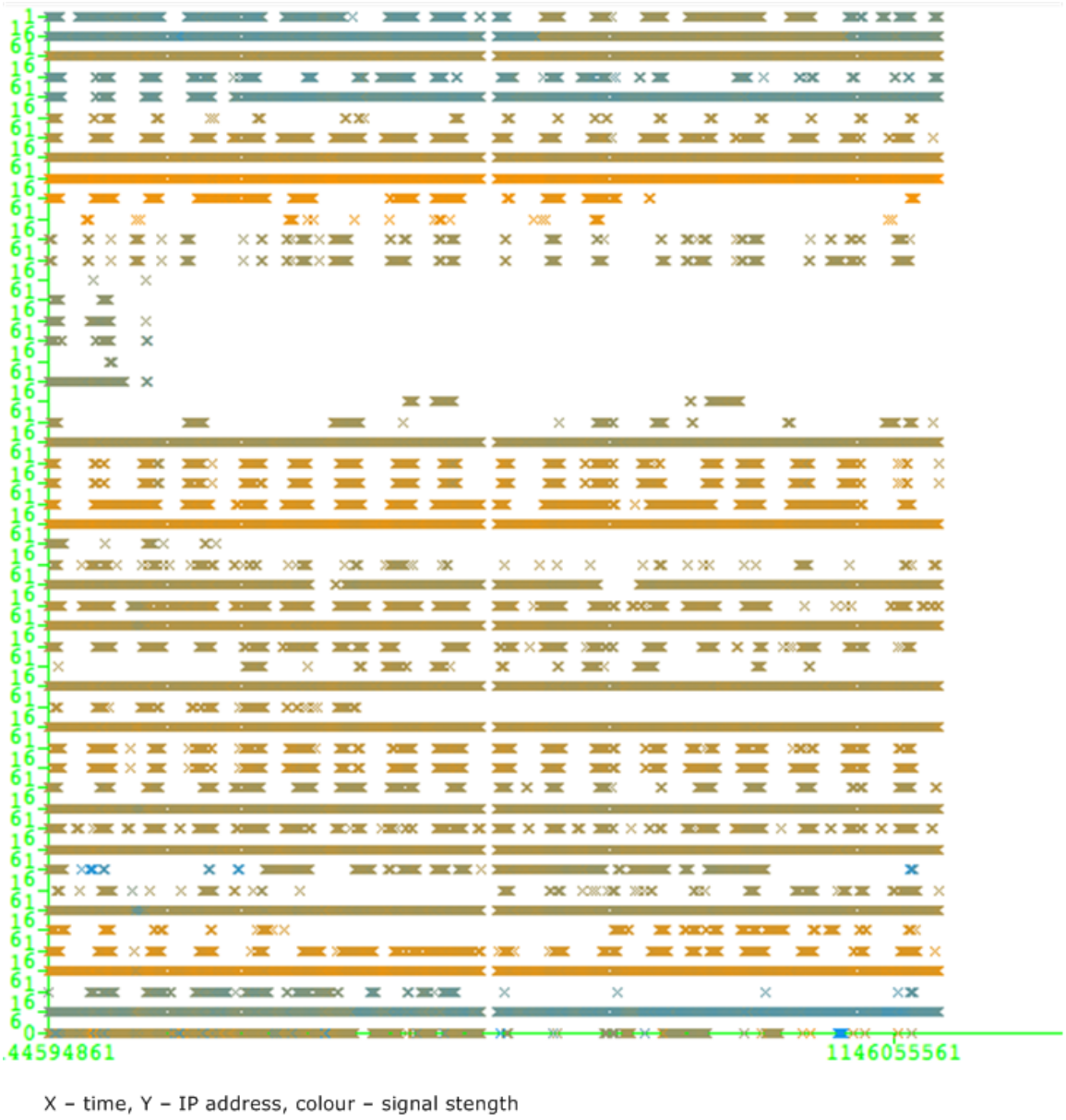
Zadavatel projektu dále požaduje, aby uživatelské rozhraní umožnilo zobrazování grafů. Grafy by mělo být možné tvořit za volitelné období. Předpokládá se možnost tvořit příslušný graf za poslední *hodinu*, *den*, *týden* a *měsíc*.

Jedním z grafů bude graf síly signálu klienta. Pokud graf za zvolené období bude obsahovat příliš mnoho hodnot na zobrazení, předpokládá se agregace dat a vykreslení průměrného signálu. Z grafu musí být patrné, v jakých intervalech jsou zobrazovaná data agregována. Graf by měl také vhodně zobrazit minimální a maximální hodnotu z agregovaných dat.

Dalším požadovaným grafem je graf přístupů klientů do sítě repuNET. Podobný graf se podařilo vygenerovat v systému Weka při dolování znalostí z dat přístupových bodů. Tento graf přehledně zobrazuje přístupy klientů do sítě ve zvoleném časovém intervalu. Graf ze systému Weka je uveden na obrázku 2.1.

Jednotlivé křížky vyznačují v daném čase přístup klienta. Barva těchto křížků zobrazuje sílu signálu klienta. Uvedený graf zobrazuje časové období jeden týden.

Graf přístupů klientů by v budoucím systému měl umožnit volit způsob výběru klientů, pro které se kreslí graf. Předpokládá se výběr klientů dle průměrné síly signálu, rozptylu hodnot síly signálu a rozdílu maximální a minimální síly signálu.



Obrázek 2.1: Graf přístupů klientů pořízený programem Weka

Další důležitou částí grafu je barvení zobrazovaných křížků. Zadavatel projektu požaduje dva druhy barvení. První způsob barvení bude barvit křížky dle absolutní hodnoty síly signálu klienta. Druhý způsob barvení, který je zajímavější z hlediska řešení bezpečnosti, bude barvit křížky dle odchylky síly signálu od dlouhodobého průměru síly signálu klienta.

2.3.2 Uživatelské účty

Jak již bylo uvedeno zadavatel projektu požaduje, aby aplikace umožňovala přihlašování uživatelů. Dalším požadavkem je možnost nastavit přístupová práva pro jednotlivé uživatele

systému. Tyto práva klientovi umožní informace z dané stránky číst nebo i pomocí ovládacích prvků tyto informace měnit. Podle toho, jestli přihlášený klient má právo měnit data, bude i uživatelské rozhraní generovat příslušné ovládací prvky.

Oponent semestrálního projektu vznesl připomínky ve smyslu, že systém by měl pravděpodobně obsahovat „logování“ chování přihlášeného uživatele. Po zvážení této připomínky jsme se zadavatelem projektu přistoupili k vytvoření uživatelského účtu *auditor*. Systém bude průběžně zaznamenávat důležité kroky přihlášeného uživatele. Tyto kroky budou zaznamenávány do logu událostí. Čtení logu událostí je umožněno pouze správci systému a auditorovi. Mazání jednotlivých položek logu bude umožněno pouze auditorovi. Auditor může ostatní stránky systému pouze číst. Tímto je oddělena moc výkonná od kontrolní autority.

2.4 Analýza dat o provozu WiFi sítí

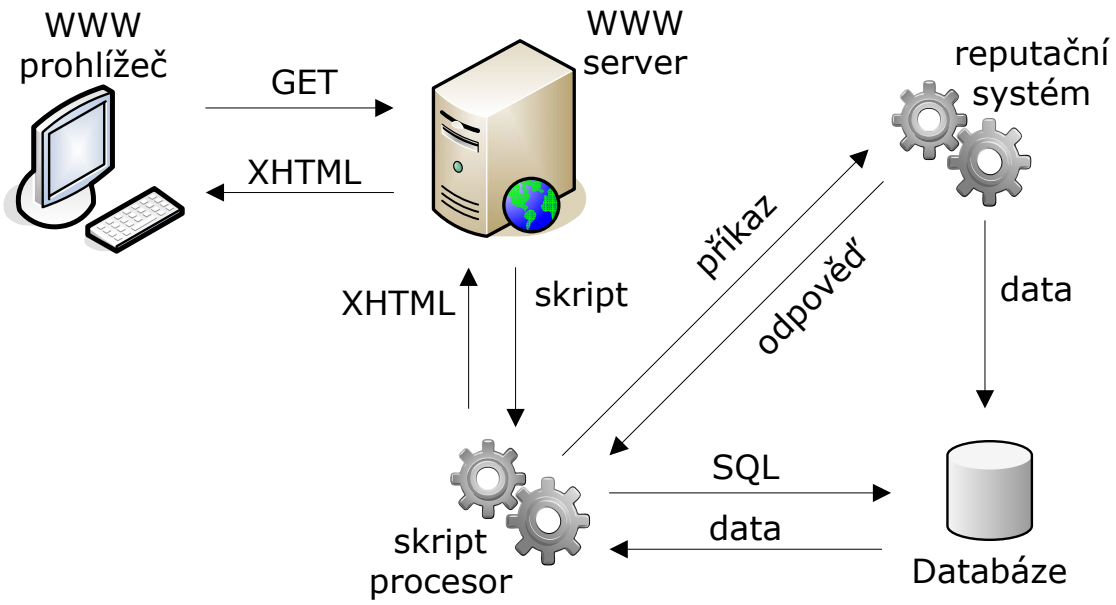
Reputační systém bude zkoušen na reálné bezdrátové síti, proto jsme se zadavatelem projektu dohodli zaměřit pouze na jeden typ přístupových bodů, který daná síť používá. V tomto případě se jedná o přístupové body založené na systému RouterOS [7].

Analýza potvrdila potřebu informací, které používala současná implementace reputačního systému. Jedná se o data přístupových bodů, zejména jejich nastavení, dále pak směrovací záznamy a registrace klientů přístupových bodů. Nově jsou použity informace o umístění přístupového bodu, které jsou zadané ve formátu WGS-84 [9]. Tyto informace budou později použity pro vygenerování mapy s topologií sítě. Použití reálných souřadnic polohy přístupových bodů by mělo umožnit ve vygenerované mapě zobrazit reálné vzdálenosti přístupových bodů.

2.5 Dostupné technologie pro řešení zadání

Požadavky systému vymezené v kapitole 2.2 předurčují využít při řešení projektu koncept internetové aplikace. Toto řešení umožní překlenout bolesti klasických aplikacích, které jsou v naprosté většině platformě závislé. Propojení s datovým úložištěm a připojením na Internet bývá v internetové aplikaci řešeno jednodušeji než u klasických aplikací.

Nespornou výhodou řešení projektu jako internetové aplikace je nenáročnost na instalovaný software uživatele systému. Uživatel může se systémem pracovat pouze pomocí internetového prohlížeče. Obecné schéma internetové aplikace rozšířené o prvky reputačního systému je zobrazeno na obrázku 2.2. Z uvedeného nákresu vyplývá, že pro řešení celého systému je třeba vyřešit způsob uložení dat v datovém úložišti, zvolit vhodný webový server a programovací jazyk. Těmto a dalším požadavkům na reputační systém věnuji pozornost v několika následujících kapitolách.



Obrázek 2.2: Architektura internetové aplikace rozšířená o reputační systém

2.5.1 Volba programovacího jazyka

V projektu je třeba zvolit jazyk, ve kterém bude naprogramována aplikace běžící na webovém serveru. Zadavatel projektu nespecifikoval programovací jazyk, ve kterém by aplikace měla být naprogramována, proto jsem si zvolil jazyk, ve kterém mám několikaletou praxi a myslím si, že je vhodný pro řešení projektu. Jedná se o skriptovací jazyk PHP verze 5. Vlastnosti jazyka, které přispěly k jeho výběru uvádím v několika následujících bodech:

- znalost a zkušenosti s jazykem programátora projektu
- jedná se o open source projekt
- cena (jedná se o free software)
- ve verzi 5 kompletně přepracována a rozšířena podpora pro objektové programování
- umožňuje pracovat s běžně používanými databázovými systémy a webovými servery
- lze provozovat na platformě Unix i Windows
- silná uživatelská základna
- dostupnost dokumentace, článků a knížek věnujících se jazyku PHP

Zmiňované informace jsem čerpal částečně ze svých zkušeností a z [10] a [11].

2.5.2 Zajištění perzistentních dat (volba databáze)

Z požadavků na datové úložiště uvedených v kapitole 2.2 vyplývá, že datové úložiště musí dostatečně zabezpečit přístup k datům a podporovat skriptovací jazyk, pomocí kterého lze převést část výpočtů do DB vrstvy.

Zadavatel projektu dále specifikoval, že objem ukládaných dat bude závislý na velikosti sítě, kde bude reputační systém nasazen. Bude se jednat o jednotky až desítky gigabajtů dat. Zadavatel projektu dále doporučuje uvážit při výběru databáze produkty MySQL, Oracle, PostgreSQL a případně další vhodné databázové systémy. Zvolený databázový systém by měl být poskytován zdarma, aby nezvyšoval celkovou cenu reputačního systému při jeho nasazení. Pro přehlednost uvádím tabulku číslo 2.1, ve které je srovnání zmiňovaných databázových systému dle klíčových vlastností pro reputační systém. Uvedené vlastnosti da-

Produkt	MySQL 5	IBM DB2 Express-C	Oracle 10g Express Edition	PostgreSQL 8
Nároky na HW	CPU alespoň PII, 100 MB RAM, 200 MB volného místa na disku	CPU alespoň 2,5 GHz, 3-4 GB RAM, Disk 100 až 500 GB	CPU od 1 GHz, 1 GB RAM, 2,5 GB volného místa na disku	CPU PIII a vyšší, 512 MB RAM, 500 MB volného místa na disku
Maximální velikost databáze	neomezeno, vázáno na OS a jeho souborový systém	omezeno jen podporovanou velikostí paměti na 4 GB	4 GB	neomezeno
Podporované operační systémy	Windows, Linux, Solaris, MacOS, NetWare	Windows, Linux	Windows, Linux	Windows, Linux, FreeBSD, MacOS, Solaris
Podpora trigerů	ano	ano	ano	ano
Podpora uložených procedur	ano	ano	ano	ano
Cena	zdarma	zdarma	zdarma	zdarma

Tabulka 2.1: Důležité parametry pro analýzu databázových systémů

tabází MySQL jsem získal z [12], DB2 z [13], Oracle z [14] a PostgreSQL z [15].

Z uvedených databází lze ihned vyřadit databázový systém firmy Oracle, protože velikost dat, které lze v tomto systému uložit, je striktně omezena na 4 GB dat. Databáze DB 2 firmy IBM má vysoké nároky na HW a při dosažení implementačních limitů této distribuce by bylo nutné migrovat na vyšší produkt jejich databáze, která již není zdarma.

Zbývající dva databázové systémy MySQL a PostgreSQL jsou open source projekty, které splňují vytyčené požadavky reputačního systému. Po konzultaci s řešitelem projektu, na který tato diplomová práce navazuje, bylo ustoupeno od použití databáze MySQL z důvodu pomalosti databázové systému, při práci s velkými objemy dat. Z toho důvodu je současná implementace reputačního systému postavena nad databází PostgreSQL, která mimo jiné umožňuje programovat uložené procedury v jazyku Python. Této vlastnosti je využito především proto, že současné implementace skriptů reputačního systému jsou napsané právě v jazyku Python.

2.5.3 Volba šablonovacího systému

Dle požadavků uvedených v kapitole 2.3 je nutné, aby uživatelské rozhraní reputačního systému bylo možné jednoduše rozšiřovat a udržovat pro další verze reputačního systému. Uživatelské rozhraní také musí podporovat jazykové mutace. Tyto požadavky směřují k využití šablonovacího systému.

Využitím šablonovacího systému lze jednoduše oddělit aplikační a prezentační logiku aplikace. Lze se tedy ve skriptech s logikou aplikace soustředit pouze na přípravu a zpracování dat pro výstup uživatelského rozhraní. Prezentaci takto připravených dat již zařídí šablonovací systém, kterému je předpřipravena šablona, dle které sestaví výstup.

Šablonovacích systémů je v dnešní době nepřeberné množství a je velice těžké určit, který bude pro projekt nejvhodnější. Na základě vlastních zkušeností jsem zvolil šablony naprogramované v jazyku *XSL*[4]. Toto řešení má výhodu v tom, že lze použít XSL transformaci pro generování téměř jakéhokoliv výstupu. Výstupem může být klasické XHTML[5], nebo např. dokument typu PDF[6]. Data pro šablonu jsou připravena ve formátu XML[7]. Využití XML a XSL je vhodné především proto, že se jedná o technologie, které jsou standardizované a jsou podporované širokou základnou programovacích jazyků. Tyto vlastnosti by měly projektu v budoucnu zajistit jednodušší rozšiřování jazykových mutací a případných změn vzhledu aplikace pouhým přepsáním šablon.

Informace týkající se XML a přidružených technologií jsou obsahově přesahující rámec této práce, proto odkazuji případné zájemce o toto téma na literaturu [16], ze které jsem většinu informací čerpal.

2.5.4 Formování obsahu dokumentů

Pro formování obsahu internetových stránek v takové podobě, v jaké je známe dnes, se již od počátku používá značkovacího jazyka *HTML*. HTML je zkratka pocházející z anglického názvu HyperText Markup Language.

[4] z anglického eXtensible Stylesheet Language - jazyk pro vytváření šablon
[5] z anglického eXtensible HyperText Markup Language - značkovací jazyk pro popis obsahu dokumentu
[6] z anglického Portable Document Format
[7] z anglického eXtensible Markup Language - univerzální rozšiřitelný značkovací jazyk

Bohužel v definici jazyka HTML se nepodařilo oddělit vrstvu definující strukturu dokumentu od vrstvy prezentační, která se používá pro formování vzhledu dokumentu. Současné internetové aplikace opouštějí tento zastaralý jazyk a objevuje se stále více aplikací, které pro formování obsahu dokumentů využívají evolučně novější jazyk XHTML. Jazyk XHTML, pro jehož definici je použit jazyk XML, bude použit pro formování obsahu stránek uživatelského rozhraní reputačního systému.

Využití tohoto jazyka je vhodné právě pro dobré oddělení prezentace dat od jejich struktury. Vhodné je také ve spojení s technologií pro formování vzhledu, kterou zmiňuji v následující kapitole. Informace zmiňované v této kapitole jsem čerpal z [17].

2.5.5 Formování vzhledu

Jak jsem již uvedl v předchozí kapitole o formování obsahu dokumentů, jazyk XHTML nemůže ovlivnit výsledný vzhled internetových stránek. Vzhled stránek naprogramovaných v tomto jazyce určují přednastavené hodnoty internetových prohlížečů. Je tedy třeba využít technologie, která takto napsaným dokumentům zajistí prezentaci. K tomuto účelu se používá jazyk CSS[8].

Využitím kaskádových stylů lze tedy odděleně definovat vzhled aplikace. Je možné využít jednoho stylu pro unifikovanou definici vzhledu celé aplikace nebo vytvořit několik alternativních vzhledů pro stejný obsah.

Pro využití kaskádových stylů jsem se rozhodl hlavně z důvodu možnosti oddělit definici prezentace do samostatných souborů. Toto velice zpřehledňuje implementované skripty aplikace a umožňuje jejich jednodušší údržbu. Informace, které zde uvádím, jsem čerpal z [18].

2.5.6 Požadavky na server systému

Zvolený koncept internetové aplikace vyžaduje pro svůj běh umístění skriptů aplikace na webový server. Je nutné, aby webový server podporoval zvolený skriptovací jazyk PHP.

Pro aplikaci byl zvolen webový server Apache. Tento server je zastoupen v drtivé většině internetových projektů. Jeho velkému rozšíření přispívá fakt, že se jedná o *open source* projekt. Server Apache je možné provozovat na platformě Unix i Windows.

Výsledný reputační systém by měl pracovat na samostatném serveru, který bude obsahovat jak zmiňovaný webový server, tak i server databázový. Toto řešení by mělo usnadnit nasazení reputačního systému v cílové síti. Umístění databázového serveru na stejný stroj není nutnou podmínkou, lze využít i existující databázový server, který již cílová organizace používá.

Současný reputační systém pracuje pouze na serveru s operačním systémem Linux. Toto omezení je především kvůli obvolávacím (senzorovým) skriptům, které jsou napsané v ja-

[8] z anglického Cascading Style Sheets - technologie kaskádových stylů

zyku Python a používají knihovnu *openssh*. Výsledný reputační systém by mohl teoreticky pracovat i na platformě Windows, bylo by ale nutné přepsat některé skripty.

Pro řešení projektu byl vyčleněn speciální server v laboratoři BUSlab[9] na FIT VUT v Brně. Konfiguraci a údržbě serveru se věnuje Petr Blahák, kterému touto cestou děkuji.

2.6 Souhrn vlastností vyvíjeného systému

- pro řešení projektu byl zvolen koncept internetové aplikace
- skripty aplikace uživatelského rozhraní budou napsány v jazyku PHP 5
- perzistenci dat zajistí databázový systém PostgreSQL
- při vývoji systému bude použit šablonovací systém využívající šablon XSL
- jazyk XHTML se využívá pro formování obsahu stránek
- formování vzhledu bude vytvořeno pomocí kaskádových stylů

[9]Brno University Security laboratory - laboratoř zabývající se bezpečností na FIT VUT v Brně

Kapitola 3

Návrh

Podrobná analýza obdobných systémů a technologií, kterou jsem popsal v kapitole 2, přispěla k přesnějšímu vymezení cíle projektu a výrazně urychlila návrh a implementaci uživatelského rozhraní reputačního systému. Důležité části týkající se návrhu aplikace a schématu databáze uvádím v této kapitole.

3.1 Organizace projektu

Na úvodní schůzce se zadavatelem projektu bylo určeno, že na projektu budu spolupracovat s původním autorem projektu Petrem Blahákem. Po zhodnocení zadání a požadavků zadavatele projektu byla práce na projektu rozdělena následovně.

Jak již bylo řečeno v 2.1.1, implementace výpočtu reputace současného systému není dostatečná. Z tohoto důvodu bylo rozhodnuto o rozšíření a upravení stávajícího reputačního systému. Petr Blahák se bude zabývat vývojem jádra reputačního systému, senzorových skriptů obstarávajících data pro reputační systém a způsobem zpětné vazby systému.

Má práce na projektu bude spočívat ve zpracování získaných dat z přístupových bodů a vytvoření uživatelského rozhraní aplikace. Data, která získají senzorové skripty, budou ukládána přímo do databáze. Zjednodušeně lze v architektuře systému moji práci na projektu označit jako práci od databáze *výše*.

Projekt tedy v současné době tvoří dva programátoři. Je tedy třeba zvolit vhodné nástroje a postupy pro týmovou spolupráci. Pro sdílení zdrojových kódů a správu verzí je pro projekt k dispozici systém CVS[1].

Komunikace mezi vedoucím projektu a mezi programátory byla z větší části omezená na využití elektronické pošty a internetových *messengerů*. Tento způsob komunikace nabyl na důležitosti především v době, kdy vedoucí projektu odcestoval do zahraničí a celý projekt řídil na dálku.

[1] z anglického Concurrent Versions System

Jako centrální místo pro nechávání vzkazů a poznámek k vývoji systému byla založena Wiki. Bohužel po havárii počítače, na kterém byl systém Wiki provozován, se nepodařilo většinu dat zachránit.

3.1.1 Model životního cyklu projektu

Jako model životního cyklu projektu byl zvolen iterativní životní cyklus s přírůstky. Tento model byl zvolen především z toho důvodu, že na projektu pracuje několik autorů a není jisté, zda všichni programátoři budou na projektu pracovat až do jeho cílové podoby. Zvolením tohoto modelu životního cyklu lze dosáhnout vytvoření určitého základu aplikace a dále ho rozšiřovat o další funkčnosti v každé iteraci.

Jednotlivé iterace končí sestavením, nebo chcete-li konstrukcí z anglického *build*. Jedná se již o funkční program, který vždy implementuje jen část celkového zadání projektu. Po každém sestavení se provádí testování a konzultace výsledku iterace se zadavatelem projektu. Tento přístup umožňuje rychlou zpětnou vazbu a zakomponování změn nebo oprav do další vývojové iterace projektu.

Pro každou iteraci vývojového cyklu projektu je stanoveno vždy několik milníků. V následujícím výčtu uvádím obsah jednotlivých milníků, které se týkají mojí práce na projektu. Tyto milníky jsou chronologicky řazeny. Uvedené milníky byly v projektu stanoveny a v současné době jsou dokončeny.

- návrh rozvržení uživatelského rozhraní a jeho barev
- převod návrhu do kaskádových stylů s optimalizacemi pro jednotlivé prohlížeče
- vložení nákresu topologie využitím formátu SVG
- vytvoření tříd pro generování stránek pomocí šablonovacího systému
- rozšíření základních tříd o jazykové mutace
- vytvoření tříd pro práci se zvoleným databázovým systémem
- tvorba formulářů pro vkládání dat do databáze
- rozšíření informací o přístupovém bodu o souřadnice ve formátu WGS84
- transformace souřadnic do formátu JTSK
- vytvoření třídy pro práci se SVG soubory
- rozšíření projektu o autentizace uživatelů
- administrace uživatelů reputačního systému
- algoritmus pro kreslení topologie sítě a jeho nasazení

- implementace tříd pro řízení práv uživatelů a stránkování výpisu dat z databáze
- rozšíření třídy pro práci se SVG o kreslení detailů přístupových bodů
- zajištění skriptů pro vytváření grafů
- vytvoření speciálních nastavitelných grafů
- vytvoření tříd pro účtování (logování činností přihlášených uživatelů) a upravení práv uživatelů
- logické rozdělení aplikace na administrační část a část týkající se bezpečnosti
- konečné úpravy grafických detailů uživatelského rozhraní a ladění stylů pro stejné zobrazení v jednotlivých prohlížečích

Téměř v každém milníku projektu byl zároveň upřesňován diagram případů použití (Use-Case) a ER[2] diagram databáze. Průběžně byla také psána diplomová práce, která dokumentuje jednotlivé kroky vývoje aplikace systému.

3.2 Návrh zpracování uživatelského rozhraní

Uživatelské rozhraní je souhrnný název pro ovládací prvky a způsob komunikace aplikace s jejím uživatelem. Nepřehledné uživatelské rozhraní nebo rozhraní s nevhodně umístěnými ovládacími prvky, může vést až k tomu, že uživatel nebude schopen danou aplikaci používat. Při návrhu uživatelského rozhraní reputačního systému jsem se snažil brát zřetel na zmiňované nedostatky některých uživatelských rozhraní.

Systém obsahuje pouze jednu hlavní nabídku, která pro přehlednost neobsahuje další úrovně zanoření jednotlivých položek. O výběru položky z hlavní nabídky je uživatel informován změnou její barvy.

Další součástí uživatelského rozhraní je panel pro volbu jazykové mutace. Změnu jazyku, kterým aplikace komunikuje, lze provést jednoduchým kliknutím na vlajku příslušného státu.

Uživatelské rozhraní reputačního systému také obsahuje nápovědu ke každé webové stránce. Tato nápověda je umístěna v pravé části obrazovky zobrazených internetových stránek. Nápověda obsahuje důležité poznámky k aktuálně zobrazené stránce nebo pouze vysvětluje zobrazené pojmy. Pokud uživatel tuto nápovědu nepotřebuje, může ji jednoduše zavřít, a zvětšit tak prostor pro prezentované informace zobrazené stránky.

Velice důležitou částí uživatelského rozhraní jsou také formuláře, které se používají pro zadávání dat. Více o formulářích, které jsou v aplikaci použity, zmiňuji v kapitole 4.

Každá stránka obsahuje nadpis. Tímto se může uživatel jednoduše orientovat, na jaké stránce se nachází. Informace o úspěšnosti prováděných akcí uživatele je vypisována

[2] z anglického Entity Relationship - diagram obsahující entity a jejich vztahy

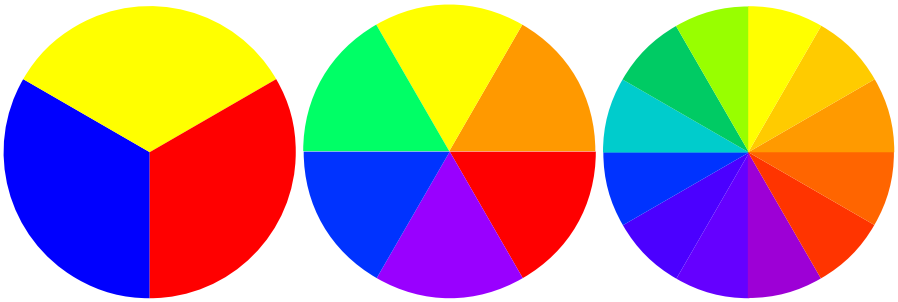
právě pod tento hlavní nadpis. Volba jednoho místa pro výpis těchto důležitých informací zpřehledňuje celé uživatelské rozhraní a usnadňuje práci uživatele.

V uživatelském rozhraní je dle analýzy třeba zajistit i řízení práv jednotlivých uživatelů. Současná implementace zobrazuje ovládací prvky pouze uživatelům, kteří pro jejich použití mají dostatečná oprávnění.

3.2.1 Volba barev a grafických prvků

Neodmyslitelnou součástí návrhu uživatelského rozhraní je také správná resp. vhodná volba barev. Jak by se na první pohled mohlo zdát, nejedná se o činnost zcela náhodnou, ba naopak. Volba barev se řídí řadou doporučení a postupů, které vznikly na základě tzv. teorie barev. Kořeny této teorie sahají až do osmnáctého století. Poznatky a fakta, které uvádím v této kapitole, jsem čerpal z [19].

Vychází se z tzv. kruhového diagramu barev. Nutno poznamenat, že kruhový diagram není jediným existujícím modelem reprezentace barevné škály. Základem diagramu jsou tzv. primární barvy, v našem případě modrá, žlutá a červená. Mícháním primárních barev vznikají barvy sekundární. Dalším mícháním sekundárních barev vznikají barvy terciální. Takto lze pokračovat, až dokud nezískáme barvy celého spektra. Kruhové diagramy obsahující primární, sekundární a terciální barvy jsou uvedeny na obrázku 3.1.



Obrázek 3.1: Kruhové diagramy barev

Snahou při návrhu barev pro uživatelské rozhraní bylo dosáhnout *harmonie* barev. Jedná se o nalezení hranice mezi malým počtem barev, které mohou působit na uživatele nudně a velkým počtem barev, které může uživatel vnímat až jako chaotické.

Při návrhu barevného schématu jsem volil barvy laděné do červeno-žluto-hnědé. Myslím si, že tato barva odpovídá tématu uživatelského rozhraní a uživateli dává dostatečně v potaz, aby zpozorněl, protože pracuje s řídícím systémem pro bezpečnost. Kromě černé a bílé jsem v návrhu volil analogické barvy. Jedná se o barvy, které jsou na zmiňovaném kruhovém diagramu sousedící. Pro jasnější zvýraznění důležitých částí uživatelského rozhraní je vždy použita barva komplementární. Tato barva je umístěna právě na opačné části kruhového diagramu, než barvy zvolené pro uživatelské rozhraní. Tímto lze jednoduše a efektivně

dosáhnout požadovaného kontrastu. Na následujícím obrázku 3.2 uvádím barevné schéma, které jsem použil pro vytvoření uživatelského rozhraní.

Obrázek 3.2: Barevné schéma uživatelského rozhraní

V uživatelském rozhraní se prolínají dvě témata. Prvním z nich jsou prvky pro zobrazení a nastavení parametrů přístupových bodů z hlediska sítí. Druhým tématem jsou prvky týkající se bezpečnosti sítě.

Pro přehledné označení příslušnosti prvků uživatelského rozhraní k těmto tématům jsem volil opět řešení pomocí barev. Prvky týkající se správy sítě jsou značeny nádechem červené barvy a obrázkem počítače. Prvky týkající bezpečnosti jsou značeny nádechem žluté barvy a obrázkem zámku. Tyto doplňující prvky uživatelského rozhraní uvádím na obrázku 3.3.

Obrázek 3.3: Označení tématických částí uživatelského rozhraní

Vytvořené uživatelské rozhraní bude dále využívat místo klasických odkazů, které často lehce splynou s okolní grafikou, zvýrazněná tlačítka. Tlačítka, mimo textu popisujícího

jejich význam, dále obsahují orámování a malou ikonu. Ikona se snaží jednoduchou grafikou oznámit uživateli o jaký typ ovládacího prvku se jedná. Vytvořená tlačítka jsou pro ukázku zobrazena na obrázku 3.4.

Obrázek 3.4: Ovládací tlačítka uživatelského rozhraní

Vzhled uživatelského rozhraní byl navržen v programu Adobe Photoshop 7.0. Prvotní návrh je uveden v dodatku B. Pro ukázku je také na obrázku B.2 uvedena finální podoba jedné ze stránek systému.

3.2.2 Návrh zpracování interaktivní mapy sítě

Mezi požadavky na uživatelské rozhraní zmiňované v kapitole 2.3 patří mj. také zobrazení topologie sítě. Po konzultaci se zadavatelem projektu byl pro zobrazení topologie využit formát SVG verze 1.1.

Zmiňovaný formát umožňuje zobrazit vektorový obraz. Toto je pro zobrazení topologie sítě velice vhodné, protože bude možné nákres topologie jednoduše zvětšovat bez ztráty kvality zobrazení.

Formát SVG dále umožňuje využít interaktivity kreslené plochy a je tedy možné definovat např. „rozkliky“ jednotlivých uzlů sítě nebo zobrazovat upřesňující informace po najetí kurzoru myši. Zmiňované a další informace o tomto formátu lze získat z [20].

Zadavatel projektu v průběhu implementace interaktivní mapy vznesl další požadavky na její funkčnost. Do návrhu zpracování interaktivní mapy sítě bylo pro další iteraci zakomponováno:

- zobrazení informací o bezdrátových rozhraních příslušného přístupového bodu po najetí kurzoru myši
- zobrazení vzdáleností mezi přístupovými body v metrech
- ovládací prvky pro posun, zvětšení a zmenšení mapy
- zobrazení alarmů týkající se problémů správy sítě a bezpečnosti k jednotlivým přístupovým bodům

3.3 Návrh uživatelského rozhraní

Pro ujasnění požadavků zadavatele projektu, při návrhu uživatelského rozhraní, jsem použil jednoduchý strukturovaný popis obrazovek. Popis obrazovky vždy obsahuje seznam prvků, které jsou na obrazovce zobrazeny. Některé obrazovky mají přesně definované i umístění jednotlivých komponent. Každá obrazovka může dále obsahovat ovládací prvky, jejichž aktivací lze přejít k zobrazení jiných obrazovek. Tyto přechody mezi obrazovkami jsou řešeny v dokumentu pomocí odkazů.

Většina stránek obsahuje pod hlavním nadpisem stránky tzv. rychlou volbu. Jedná se o roletovou nabídku, která umožňuje zvolit jiný obsah stránky v daném kontextu. Jedná se např. při výpisu stránky s detailem přístupového bodu o změnu právě vybraného přístupového bodu. Tímto lze v tomto případě elegantně přecházet mezi dalšími přístupovými body, aniž by bylo nutné vracet se na předchozí stránku.

Téměř každá stránka obsahuje prvek pro návrat na předchozí obrazovku. Tento prvek není uváděn v seznamu možných akcí obrazovek. Každá obrazovka také obsahuje ve své levé části hlavní nabídku. Strukturovaný popis obrazovek také neobsahuje obsah nápovědy. Nápověda vždy obsahuje text, který vysvětluje nebo doplňuje právě zobrazenou stránku.

V následujícím seznamu uvádím jednotlivé položky hlavní nabídky s odkazem na obrazovku, která se zobrazí při jejich aktivaci:

- Úvod → 3.3.2 Úvodní obrazovka
- Přístupový bod → 3.3.10 Výpis přístupových bodů
- Grafy → 3.3.11 Grafy
- Uživatelé → 3.3.12 Uživatelé
- Log→ 3.3.14 Log

3.3.1 Přihlašovací obrazovka

Prvky

- přihlašovací formulář (možnost zadat přihlašovací jméno a heslo uživatele)

Akce

- odeslání formuláře → 3.3.2 Úvodní obrazovka

3.3.2 Úvodní obrazovka

Prvky

- vykreslení přístupových bodů sítě dle zadaných souřadnic
- vykreslení alarmů (správa a bezpečnost)
- vykreslení spojů mezi AP dle směrovacích záznamů
- po najetí myší nad AP zobrazovat podrobnější informace o AP (detail o bezdrátových rozhraních, vzdálenosti přístupových bodů)
- ovládací prvky pro posuny a změnu velikosti vykreslené topologie sítě
- celkový počet uživatelů sítě

Akce

- kliknutí na přístupový bod → 3.3.3 Detail přístupového bodu
- kliknutí na počet uživatelů→ 3.3.10 Výpis přístupových bodů

3.3.3 Detail přístupového bodu

Prvky

- souhrnný výpis parametrů přístupového bodu včetně aktuálního počtu klientů
- administrační část
  - detailní informace o přístupovém bodu
  - aktuální počet klientů a počet klientů za poslední den, týden a měsíc
  - rozhraní přístupových bodů
- bezpečnostní část
  - seznam pěti klientů s nejhorší reputací

Akce

- úprava údajů přístupového bodu → 3.3.4 Editace detailu přístupového bodu
- zobrazení detailu rozhraní přístupového bodu → 3.3.5 Karta přístupového bodu
- zobrazení směrovacích záznamů rozhraní přístupového bodu → 3.3.5 Směrovací údaje
- zobrazení adres rozhraní přístupového bodu → 3.3.5 Adresy
- zobrazení registrací klientů přístupového bodu → 3.3.6 Registrace klientů

3.3.4 Editace detailu přístupového bodu

Prvky

- formulář umožňující změnu údajů přístupového bodu

Akce

- uložit změny → 3.3.4 Editace detailu přístupového bodu

3.3.5 Karta přístupového bodu

Prvky

- podrobný výpis parametrů rozhraní přístupového bodu, členěný dle témat do záložek
- kotva výpis IP adres rozhraní přístupového bodu
- kotva výpis směrovacích záznamů rozhraní přístupového bodu

3.3.6 Registrace klientů

Prvky

- výpis klientů registrovaných na přístupovém bodu

Akce

- zobrazit graf síly signálu klienta → 3.3.8 Graf síly signálu klienta

3.3.7 Nastavení pro vytvoření grafu přístupů klientů

Prvky

- formulář umožňující zadat časové rozpětí grafu, způsob výběru klientů, jejich řazení a barvení grafu

Akce

- odeslat → 3.3.9 Graf přístupů klientů do sítě

3.3.8 Graf síly signálu klienta

Prvky

- graf síly signálu vybraného klienta sítě v závislosti na čase

Akce

- formulář pro změnu časového intervalu osy X→ 3.3.8 Graf síly signálu klienta

3.3.9 Graf přístupů klientů do sítě

Prvky

- graf přístupů klientů do sítě repuNET

Akce

- nastavení pro vytvoření grafu→ 3.3.7 Nastavení pro vytvoření grafu přístupů klientů

3.3.10 Výpis přístupových bodů

Prvky

- tabulkový výpis přístupových bodů systému

Akce

- zobrazit detail → 3.3.3 Detail přístupového bodu
- smazat → potvrzovací dialog pro smazání přístupového bodu
- vložit nový přístupový bod→ formulář pro vložení nového přístupového bodu

3.3.11 Grafy

Prvky

- hlavní rozcestník pro tvorbu grafů

Akce

- zobrazit graf síly signálu→ 3.3.8 Graf síly signálu
- zobrazit graf přístupů klienta→ 3.3.7 Nastavení pro vytvoření grafu přístupů klienta

3.3.12 Uživatelé systému repuNET

Pruky

- tabulkový výpis uživatelů definovaných v systému (uživatelé, kteří mají přístup k uživatelskému rozhraní systému)
- vizuální rozlišení uživatelů na běžné a uživatele administrátory

Akce

- změna oprávnění uživatele → 3.3.13 Oprávnění
- smazat → potvrzovací dialog pro smazání uživatele systému
- aktivita uživatele → nástroj na zakázání nebo povolení přístupu daného uživatele
- vytvořit nového uživatele → formulář pro vytvoření nového uživatele

3.3.13 Nastavení oprávnění uživatele

Pruky

- prvek pro výběr šablony s oprávněním
- výpis aktuálně nastavených oprávnění s možností je měnit

Akce

- odeslat → 3.3.13 Oprávnění

3.3.14 Log

Pruky

- tabulkový výpis logu událostí

Akce

- smazat → potvrzovací dialog pro smazání vybrané události z logu

3.4 Návrh databáze

Při návrhu databáze bylo využito konceptuálního modelování. Výsledek modelování je zakreslen ve formě ER diagramu. Zmiňovaný model je třeba doplnit modelem tzv. funkčního modelování. Jedná se o model případů použití, který umožní zachytit operace, které mohou s daty probíhat.

Jak již bylo uvedeno v kapitole 3.1, návrh databáze se mění postupným upřesňováním zadání a rozšířeními dalších konstrukcí projektu. Z tohoto důvodu jsou diagramy uvedené v následujících podkapitolách vypovídající pouze o stavu současné implementace projektu.

3.4.1 Diagram případů použití

Model případů použití je zobrazen na obrázku 3.5. Je možné vidět, že nepřihlášený uživatel má možnost pouze změnit jazykovou mutaci prostředí a přihlásit se do systému.

Po přihlášení uživatele jsou načtena práva uživatele. Systém umožňuje měnit přístupová práva k jednotlivým stránkám uživatelského rozhraní. Uživateli lze přidělit právo *čtení* a právo *změn*. Tato práva může určit pouze administrátor systému.

Diagram případů použití uvádí jednotlivé akce z pohledu uživatele, který má právo čtení a uživatele, který má právo změn na všechny stránky systému. Jednotlivé případy použití dále nekomentuji, protože jejich význam je zřejmý z jejich pojmenování.

Dále v systému může existovat uživatel, který má nastavená oprávnění dle šablony *auditor*. Tento typ uživatele má právo číst veškeré stránky systému, ale může měnit pouze stránku s výpisem událostí.

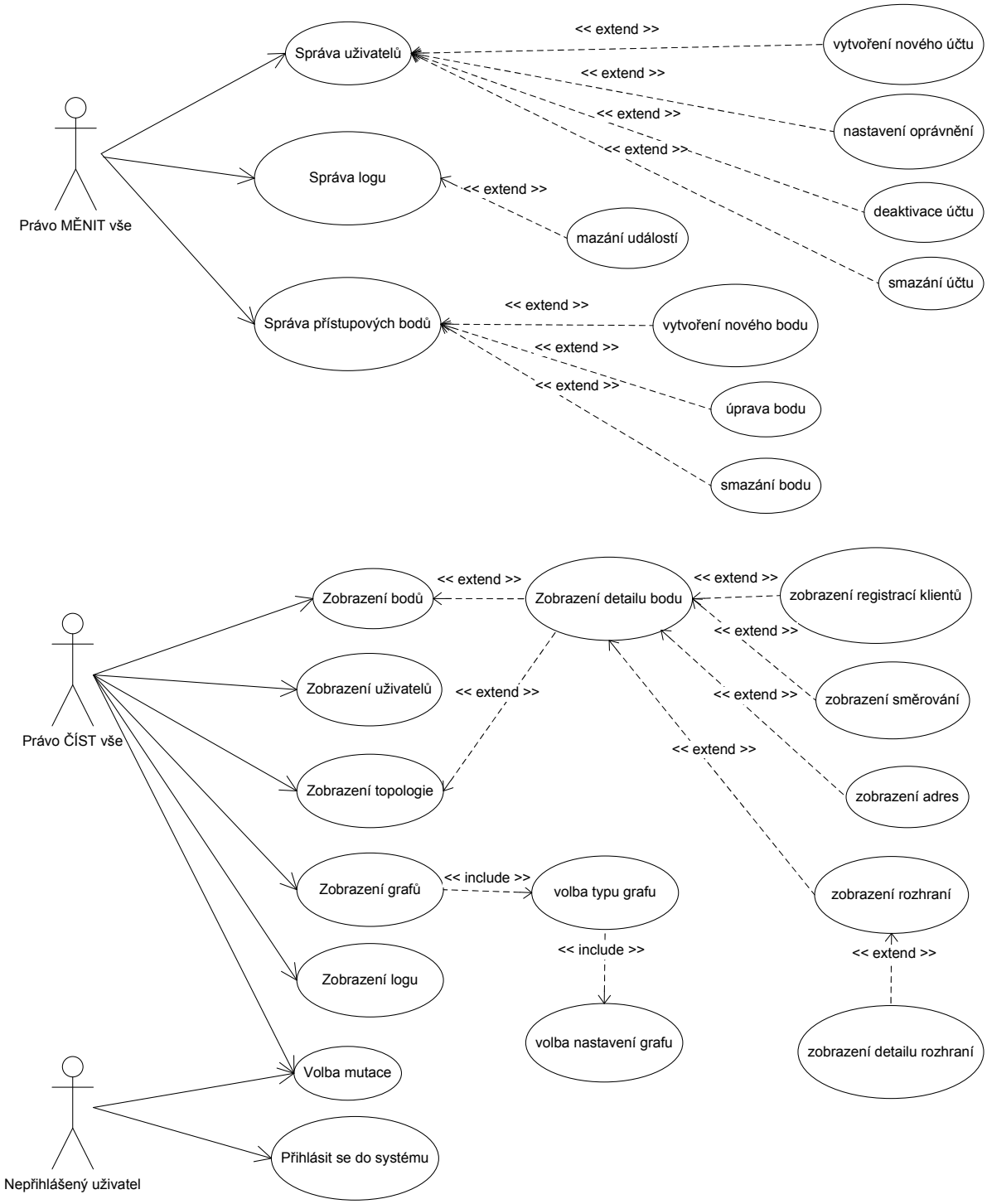
3.4.2 ER Diagram

Pro vytvoření ER diagramu jsem použil informace, které jsem získal v průběhu analýzy projektu. Současný ER diagram obsahuje 9 entit. Entity *„Accesspoint“*, *„Address“*, *„Registration“*, *„Route“* a *„WLan“* vznikaly dle požadavků skriptů, které získávají data z přístupových bodů. Atributy těchto entit odpovídají datům, které poskytují senzorové skripty.

Volba takovéto struktury umožňuje zrychlení práce senzorových skriptů, které nemusejí získaná data transformovat. Rychlost senzorových skriptů reputačního systému je jedním z požadavků plynoucích z analýzy a formát těchto entit tedy bude zachován i na úkor vzniku redundance dat ve vytvořených tabulkách.

Zakomponováním funkčních požadavků plynoucích z vytvořeného diagramu případů použití vznikl výsledný ER diagram, který je zobrazen na obrázku 3.6.

Entita *„User“* obsahuje informace o uživateli systému. Při nastavení oprávnění uživatele lze použít šablonu, která obsahuje přednastavená oprávnění. Šablony, ke kterým se vážou jednotlivá oprávnění jsou uloženy v entitě *„UserType“*.



Obrázek 3.5: Diagram případů použití

Pokud jsou práva uživateli přidělena šablonou, je využit vztah *„je typu“*. Pokud správce uživatelských účtů nevyužije šablony, ale definuje jednotlivá oprávnění individuálně, je využito vztahu „*má oprávnění*“, který je vázán k entitě „*Permission*“.

Entita *„Permission“* umožňuje uchovat informace o právech na *změnu* nebo *čtení* jednotlivých stránek uživatelského rozhraní.

Mezi klíčové entity pro uchování dat o síti patří entita *„Accesspoint“*, která umožňuje uložení informací o přístupových bodech sítě. K této entitě se vážou další entity, které ukládají informace o IP adresách, směrovacích záznamech, registracích klientů jednotlivých přístupových bodů a bezdrátových rozhraních. Neuvádím již podrobnější popis těchto entit, protože označení entit a jejich atributů odpovídá zavedeným konvencím z oblasti počítačových sítí.

Entita *„WLan“* zobrazená v ER diagramu neobsahuje výpis všech svých atributů. K tomuto kroku jsem přistoupil především pro zjednodušení výsledného diagramu. Ve skutečnosti tato entita obsahuje 48 atributů, které se týkají nastavení bezdrátových rozhraní přístupového bodu. Pojmenování atributů lze naleznout na přiloženém CD ve skriptu, který vytváří schéma databáze.

Poslední entitou je entita *„Log“*, která uchovává informace o důležitých událostech, které vnikají při práci s uživatelským rozhraním.

3.4.3 Databázové schéma

Transformací navrženého ER diagramu na schéma databáze vzniklo devět tabulek, které odpovídají jednotlivým entitám ER modelu. Stejný počet entit a vzniklých tabulek lze odůvodnit tím, že v modelu není nikde použit vztah M:N, který by vyžadoval vznik další vazební tabulky.

U výsledného schématu databáze nebyla zjišťována normální forma, které toto schéma odpovídá. Určení normální formy a případné další optimalizace databázového schématu budou provedeny až nad schématem, které bude použito v *„ostré“* verzi systému. Jak již bylo zmíněno v předchozím textu, některé redundance dat jsou ponechána úmyslně, aby senzorové skripty nemusely data složitě transformovat a upravovat.

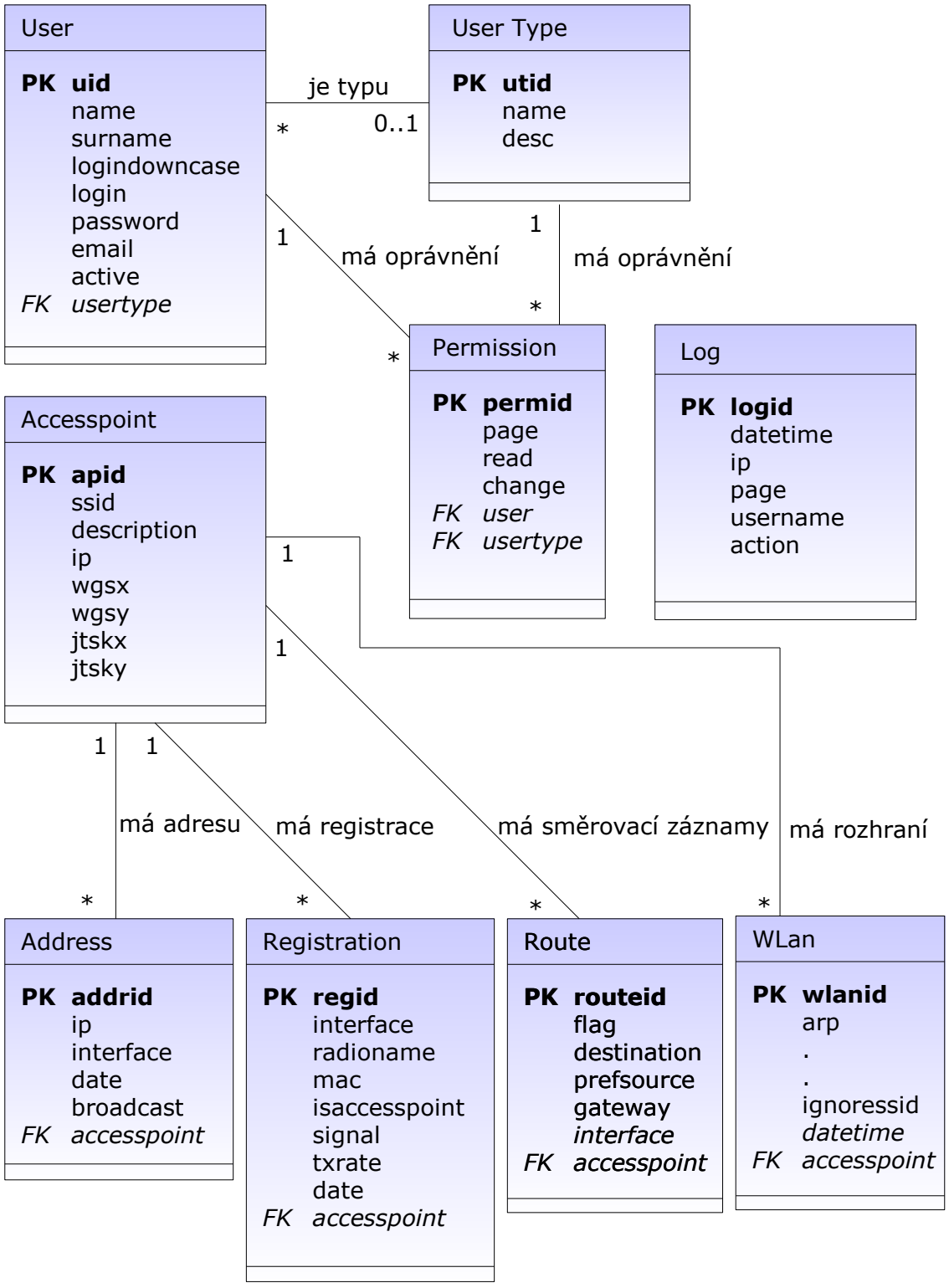
3.5 Návrh způsobu přihlašování

Návrhem způsobu přihlašování v internetových aplikacích jsem se podrobně zabýval ve své bakalářské práci [21], ze které jsem tuto kapitolu převzal.

Přihlašováním je myšlen mechanismus, který poskytne možnost ověření uživatele a dle jeho oprávnění mu umožní přístup na místa systému, které jsou nepřihlášenému uživateli nepřístupná.

Vytvoření přihlašování na internetových stránkách přináší dva základní problémy. Prvním problémem je nutnost získat přihlašovací informace od uživatele a ověřit jejich správnost.

Druhý problém je daleko složitější, je totiž třeba zajistit přenos informace o tom, zda je uživatel přihlášen mezi jednotlivými internetovými stránkami. V následujících odstavcích



Obrázek 3.6: ER Diagram

uvedu několik alternativ, které jsem zvažoval při návrhu způsobu přihlašování v mojí aplikaci a odůvodním výběr mého řešení.

3.5.1 Přihlašování pomocí JavaSriptu

S tímto způsobem přihlašování se dnes již téměř nesetkáme. Ověření hesla popř. jiných uživatelem zadaných informací se provádí pomocí kódu v jazyce JavaScript. Nevýhoda tohoto způsobu je především v tom, že kód pro ověření informací je zapsán přímo v kódu internetové stránky. Je zřejmé, že zkušenější uživatel může ze zdrojového souboru internetové stránky lehce získat heslo a jiné citlivé informace. S přihlédnutím k těmto nevýhodám se nebudu této variantě dále věnovat.

3.5.2 Využití Cookies

Co to jsou *cookies*? Překlad cookies je *„koláčky“*. Dle mého názoru je toto pojmenování zavádějící a dále v textu budu používat původní anglický název cookies. Cookie je název pro krátkou informaci, kterou může skript uložit do klientského počítače. Tyto cookie se na klientský počítač ukládají ve formě krátkých textových souborů.

Řešit problém s přihlašováním pouze pomocí cookies lze, ale přináší několik nepříjemných vlastností. Jedním z hlavních problémů je to, že některé prohlížeče cookies vůbec nepodporují, nebo je mají zakázané. Další nepříjemnou vlastností tohoto typu přihlašování bývá automatické přihlášení uživatele při vstupu na tyto stránky. Toto může být nevhodné v případě, že počítač, na kterém jsou cookies vytvořeny, používá více uživatelů. Pro svůj projekt jsem cookies využil nepřímo, více se k tomu zmíním v části zabývající se implementací projektu.

3.5.3 Využití HTTP Autentizace

Jedná se o vestavěnou podporu pro přihlašování v internetových prohlížečích a webových serverech. Po zaslání speciální hlavičky internetovému prohlížeči se zobrazí systémové okno s požadavkem na vyplnění přihlašovacích údajů. Po odeslání přihlašovacích údajů již prohlížeč sám s každou další žádostí o novou internetovou stránku posílá i tyto vyplněné informace.

Toto je první velký problém, protože data jsou v základním módu posílána jako čistý text a nelze tedy zaručit, že nebudou odposlechnuta. Je také zbytečné tato citlivá data přenášet při každé žádosti o novou stránku. Hlavně díky této vlastnosti se tento typ přihlašování používá pouze v intranetových aplikacích, kde nejsou tak vysoké nároky na zabezpečení.

Dalším problémem tohoto řešení je nelehká implementace odhlášení uživatele. Prohlížeč zapamatované údaje zapomene až při svém ukončení. Toto je problém, který se řeší pomocí JavaScritu a je třeba ho řešit individuálně dle typu internetového prohlížeče.

Další problémy vznikají s děděním těchto údajů do nových oken prohlížeče, pak se může stát, že se uživatel v jednom okně odhlásí a v ostatních je stále přihlášen.

Posledním problémem je nutnost běhu preprocesoru PHP jako modulu webového serveru. Využití tohoto přihlašování je proto na servery, kde PHP neběží jako modul, nepřenosné. I přes některé výhody tohoto řešení jsem se rozhodl použít až následující řešení.

3.5.4 Využití SESSION proměnných

Session proměnné jsou proměnné, které jsou umístěny na webovém serveru většinou ve formě souborů. Do těchto souborů je možné ukládat data, která jsou pevně svázána s jednotlivými žádostmi klientů o internetové stránky.

Důležité je, že server sám rozpozná, při každé žádosti o novou stránku, o jakého klienta se jedná a zpřístupní data v těchto proměnných náležící tomuto klientovi. Tímto mechanismem je možné sledovat uživatelův pohyb po stránkách a udržovat mj. i informaci o tom, zda je přihlášen nebo jaká má oprávnění.

Jakákoliv další funkční podpora pro řešení přihlašování uživatele již logicky není problematikou session proměnných. V mém případě to znamená, že sběr informací od uživatele a případnou logiku přihlašovacího aparátu budu muset naprogramovat sám. Jedná se o poměrně pracné řešení, ale jako autor získám absolutní kontrolu nad přihlašovacím mechanismem a mohu ho upravit dle vlastních požadavků.

Ve svém řešení přihlašování uživatelů využiji session proměnné. Podrobnější informace o přihlašovací části mojí aplikace uvádím v části věnující se implementaci.

3.5.5 Profesionální řešení přihlašování uživatele

Samozřejmě jsem nevyčerpal všechny možnosti, jak řešit přihlašování uživatele v internetových aplikacích, což také nebylo cílem. Profesionální aplikace, které kladou požadavky hlavně na bezpečnost, využívají především zabezpečený protokol *„HTTPS“*. V profesionálních aplikacích lze nalézt kombinace uvedených postupů, optimalizovaných pro jednotlivé prohlížeče. V konečné verzi reputačního systému bude využito zabezpečení pomocí zmiňovaného protokolu *HTTPS*.

Kapitola 4

Implementace

V této kapitole jsou popsány pouze důležité nebo zajímavé části implementace uživatelského rozhraní reputačního systému. Popis všech implementovaných částí systému by několikanásobně přesáhl požadovaný rozsah diplomové práce.

4.1 Odstínění obsahu a formy prezentace

Pro vytvoření uživatelského rozhraní reputačního systému jsem použil šablonovací systém, který jsem uvedl v kapitole 2.5.3. V této kapitole podrobněji popíši implementaci, která se týká tohoto šablonovacího systému.

Veškeré PHP skripty reputačního systému implementují pouze logiku aplikace. V těchto skriptech dochází k vyhodnocování podmínek, zpracování dat nebo přípravy dat pro výstup. Skripty také zajišťují kontext mezi jednotlivými stránkami přihlášeného uživatele a řídí jeho oprávnění v systému. Jakmile jsou všechna data pro sestavovanou internetovou stránku připravena, vytvoří se XML dokument, který tato data přenese. Vytvořený XML dokument je vytvořen pomocí *DOM*[1] metod jazyka PHP. Tento dokument není nikde fyzicky uložen a je pouze v paměti webového serveru.

Pro zobrazení výsledné internetové stránky je zapotřebí definovat šablonu XSL. Vytvořením vhodné šablony lze získat výstup v různých formátech a nemusíme se tedy omezovat pouze na formát internetové stránky, který bývá zpravidla ve formátu XHTML.

Vytvořenou XSL šablonu a XML dokument použiji jako vstup pro XSLT procesor, který je také součástí jazyka PHP. Tento procesor provede XLS transformaci XML dokumentu a zobrazí výslednou internetovou stránku.

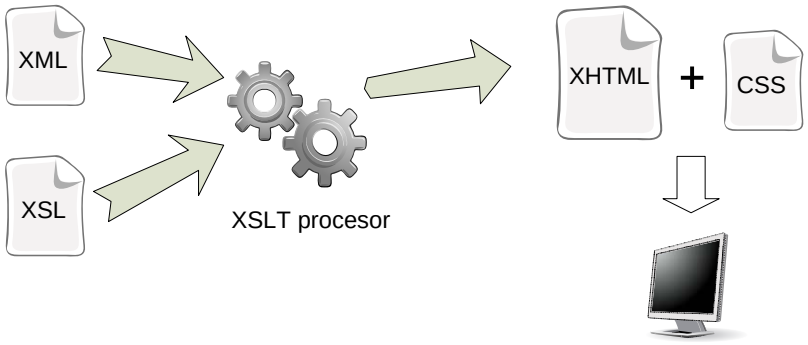
Uživatelské rozhraní reputačního systému používá pro úpravu vzhledu výsledného XHTML dokumentu kaskádových stylů. Tyto styly jsou připojeny jako externí soubor již v šablonách XSL.

Zvolený systém může na první pohled vypadat velice složitě. Složitost je ale vykoupena jednodušší údržbou celého programu. Lze upravovat jednotlivé vrstvy aplikace, aniž

[1] z anglického Document Object Model - objektový model struktury dokumentu

by bylo nutné zasahovat do vrstev sousedních. Toto v praxi znamená, že např. můžeme jednoduše změnit vzhled aplikace přepsáním šablon, aniž bychom měnili datovou vrstvu, která připravuje data pro stránku.

Celý systém od zpracování XML dokumentu až po sestavení výsledné internetové stránky ilustruji na obrázku 4.1.



Obrázek 4.1: Využití XSL šablon pro sestavení internetové stránky

4.2 Využití SVG pro interaktivní mapu

V analýze podobných produktů pro monitorování sítě zmiňuji vhodnost použití nákresu topologie sítě. Zobrazení topologie sítě podá ucelený obraz o síti reputačního systému. Topologie je kreslena mezi jednotlivými přístupovými body sítě. Pro kreslení čar mezi přístupovými body jsou použity směrovací záznamy z přístupových bodů.

Pro vlastní zobrazení nákresu topologie byl použit formát SVG. Jedná se o vektorový obraz a mělo by být možné ho jednoduše zvětšovat. Pro zápis příkazů jazyka SVG se používá jazyk XML. Zejména vektorovost formátu a možnost vytvářet obraz pomocí DOMu přispěla k volbě tohoto řešení.

Pro vytváření SVG souboru s topologií sítě jsem naprogramoval třídu *„svgpainter.class.php“*. Tato třída používá DOM funkce pomocí kterých dokáže vytvořit SVG dokument. Třída obsahuje metody pro kreslení přístupových bodů, jejich propojení, informací o přístupových bodech a ovládacích prvků pro posun a zvětšování mapy.

Největší problém jsem zaznamenal v podpoře internetových prohlížečů při zobrazení SVG dokumentu jako součásti XHTML stránky tzv. *inline* zobrazení. Požadavky prohlížeče Internet Explorer a např. prohlížeče FireFox se liší natolik, že nelze naleznout univerzální řešení bez porušení validity XHTML kódu. Pro řešení tohoto problému jsem využil článku [1].

SVG dokument vkládám do XHTML stránky pomocí značky `<embed>`, která již není do standardu XHTML zařazena. Využití této značky poruší validitu XHTML kódu, ale

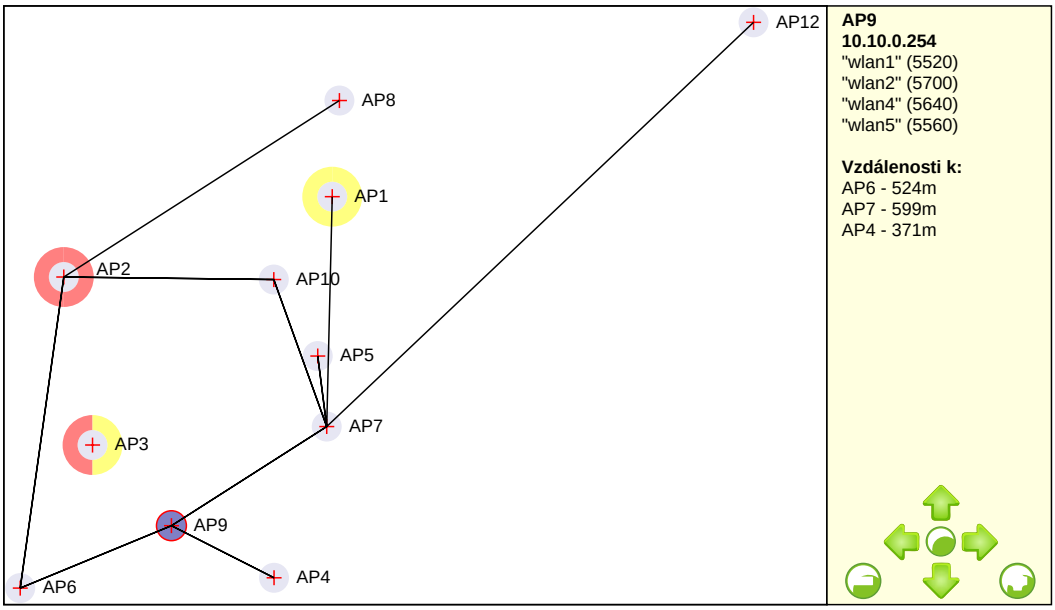
vložený SVG dokument bude správně zobrazen v prohlížečích Internet Explorer, FireFox, Mozilla a Opera.

Pro úplnost je třeba poznamenat, že prohlížeče Internet Explorer používají pro zobrazení SVG dokumentu zásuvný modul ASV 3[2] firmy Adobe, který kompletně implementuje standard SVG.

Nelze také nezmínit problémy při programování interaktivních prvků mapy. V podstatě jediný způsob, jak lze v SVG dokumentu přiřadit nakresleným prvkům nějakou akci, je takový, který vede k využití skriptovacího jazyka ECMAScript. Tento jazyk vychází z jazyku JavaScript a interaktivity se dosahuje reagováním na události, které vznikají např. při pohybu myší nebo stisku kláves na klávesnici.

Problém většinou nastává díky nestandardnímu DOMu internetové stránky v internetovém prohlížeči Internet Explorer. Je tedy většinou nutné ECMAScript kód psát minimálně ve dvou provedeních. Jednou pro Internet Explorer a podruhé pro ostatní prohlížeče.

Protože v době odevzdání diplomové práce stále není kompletní implementace reputačního systému, zobrazení alarmů v interaktivní mapě je pouze ukázkové. Tímto demonstruji, že uživatelské rozhraní je na požadované zobrazování alarmů připraveno. Výsledek implementace interaktivní mapy je zobrazen na obrázku 4.2.



Obrázek 4.2: Interaktivní mapa s topologií sítě

[2] Adobe SVG Viewer

4.3 Grafy

Pro implementaci grafů jsem se rozhodl použít formátu SVG. Snažil jsem se naleznout knihovnu, která by se dala využít při kreslení grafů. Bohužel požadavky na grafy reputačního systému, které uvádím v analýze, jsou natolik specifické, že nebylo možné využít již hotového řešení některého z nástrojů pro tvorbu grafů.

Nakonec jsem použil alfa verzi knihovny SVG Graph od pana Hermana Veluwenkampa, kterému touto cestou děkuji. Knihovna umožňuje kreslit jen jednoduché typy grafů a v současné verzi je velice jednoduchá. Z této knihovny jsem využil pouze několik funkcí, které zajišťují výpočet rozsahu hodnot a kreslení os grafu. Samotné vykreslení grafu a úpravy dat včetně jejich agregace, jsem řešil již naprogramováním vlastních funkcí.

Při průběžném testování vznikl problém s velikostí výsledného grafu. Některé SVG soubory s grafy přesahovaly velikost i stovek kilobajtů. Takto velké soubory při pomalém připojení k internetu mohou znamenat nepříjemné čekání na odezvu uživatelského rozhraní. Problém byl vyřešen použitím komprese SVG souborů do souborů SVGZ. Tyto soubory mají výslednou velikost i více jak 10 krát menší. S výhodou lze také využít toho, že veškeré testované prohlížeče podporují zobrazení i takto komprimovaných souborů.

Je vhodné poznamenat, že pokud by měl výsledný SVG soubor velikost v řádech megabajtů, nastává problém s prohlížením jeho obsahu v internetových prohlížečích. Takto obrovské grafy ale současný systém negeneruje. Nejrychleji a nejspolehlivěji soubory SVG zobrazuje prohlížeč Opera. V prohlížeči FireFox se při velkých souborech stává, že prohlížeč i na několik minut přestane reagovat. Malé soubory a tedy i grafy implementovaného systému všechny zmiňované prohlížeče, včetně Internet Exploreru, zobrazují správně.

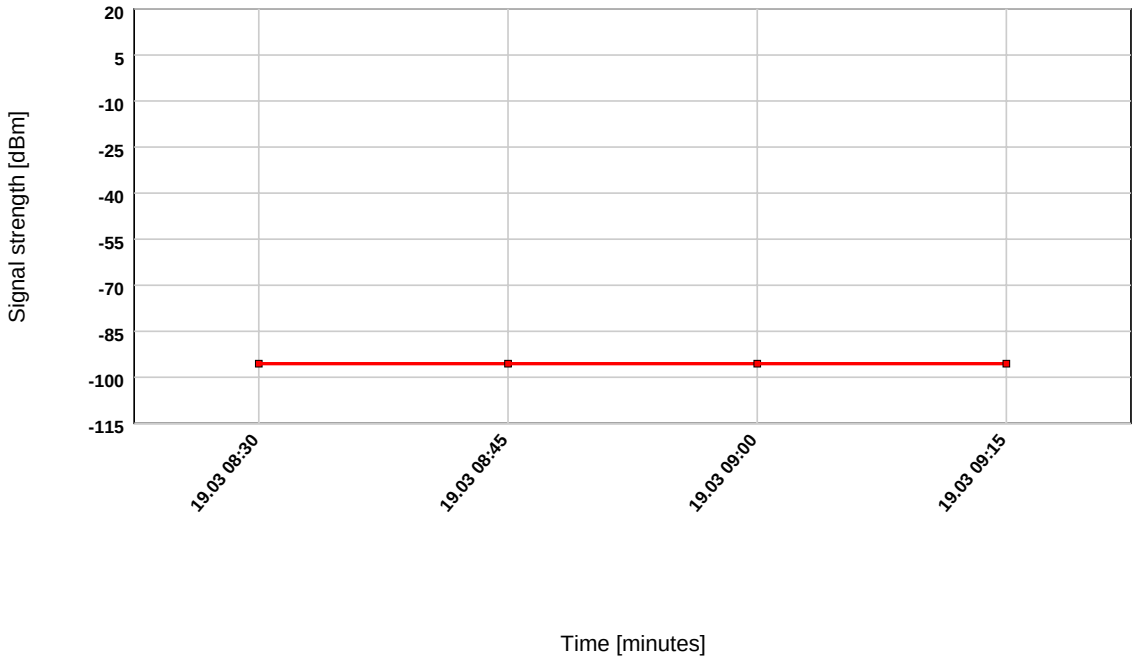
4.3.1 Graf síly signálu klienta

Graf síly signálu využívá upravený původní spojnicový graf, který je obsažen v použité knihovně. Graf byl dále rozšířen o možnost agregace dat. Pokud je grafu posláno více jak 32 hodnot, provede se agregace dat. Hodnota 32 vznikla experimentálně a dovoluje zobrazit graf s přijatelnou přehledností a výrazovou bohatostí. Pro ukázku je na obrázku 4.3 zobrazen graf síly signálu klienta bez použití agregace dat.

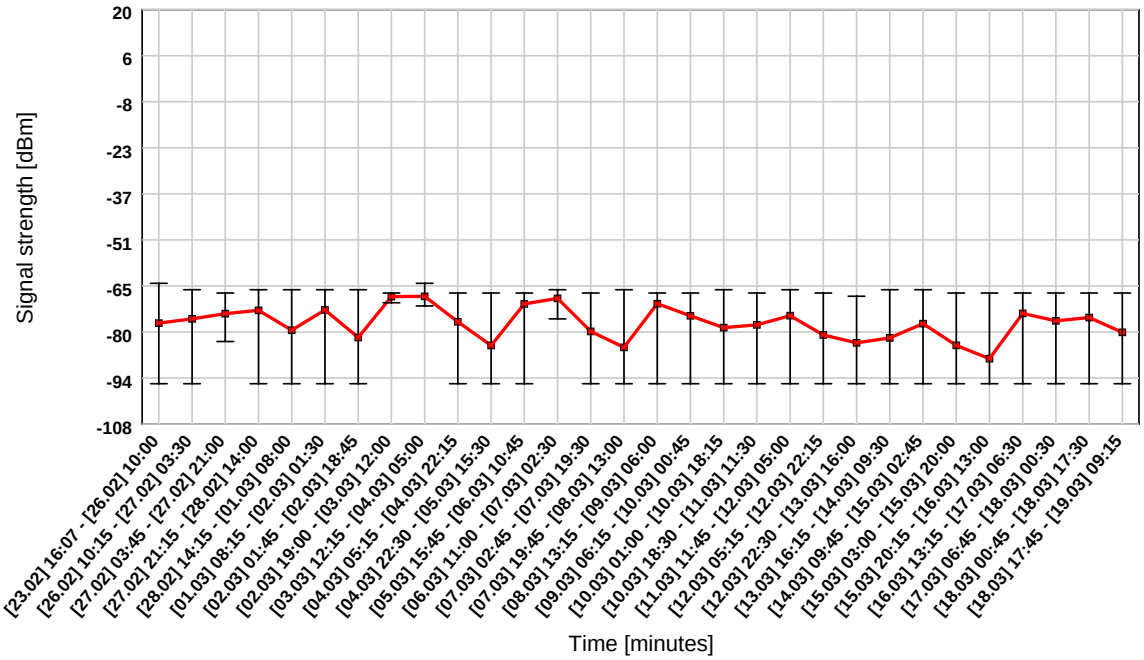
Pokud se data agregují je vypočten počet intervalů tak, aby ve výsledném grafu bylo maximálně 32 hodnot na ose x. V jednotlivých intervalech se dále počítá minimální a maximální hodnota z daného intervalu a ta se do grafu zobrazuje pomocí černých úseček. Popisek osy x v případě agregace dat obsahuje jednotlivé časové intervaly. Tento typ grafu je uveden na obrázku 4.4.

4.3.2 Graf přístupů klientů do sítě

Tento typ grafu obsahuje velice mnoho křížků, které reprezentují to, zda daný klient v daném čase využíval síť, či nikoliv. Bylo tedy nutné, aby výsledný graf byl přehledný a posloužil



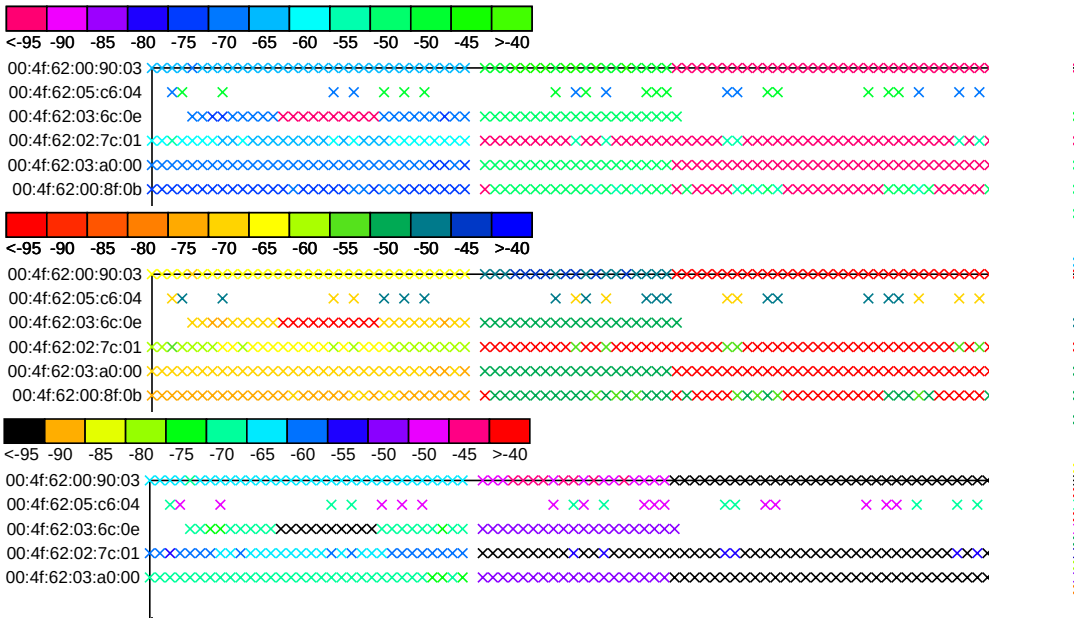
Obrázek 4.3: Graf síly signálu klienta bez agregace hodnot časové osy (interval 1 den)



Obrázek 4.4: Graf síly signálu klienta s agregací hodnot časové osy (interval 1 měsíc)

tak svému účelu. Jedinou možností bylo zvolení vhodného obarvení grafu, které bylo prove-

deno experimentálně. Několik zvažovaných variant, které jsem konzultoval se zadavatelem projektu, je uvedeno na obrázku 4.5.



Obrázek 4.5: Varianty grafu přístupů klientů (volba barev)

První varianta grafu využívala barev celého spektra vyjma barev laděných do žluté. Vypuštěním žlutých barev jsem si sliboval lepší čitelnost grafu, protože žlutá barva na bílém podkladu dobře zaniká.

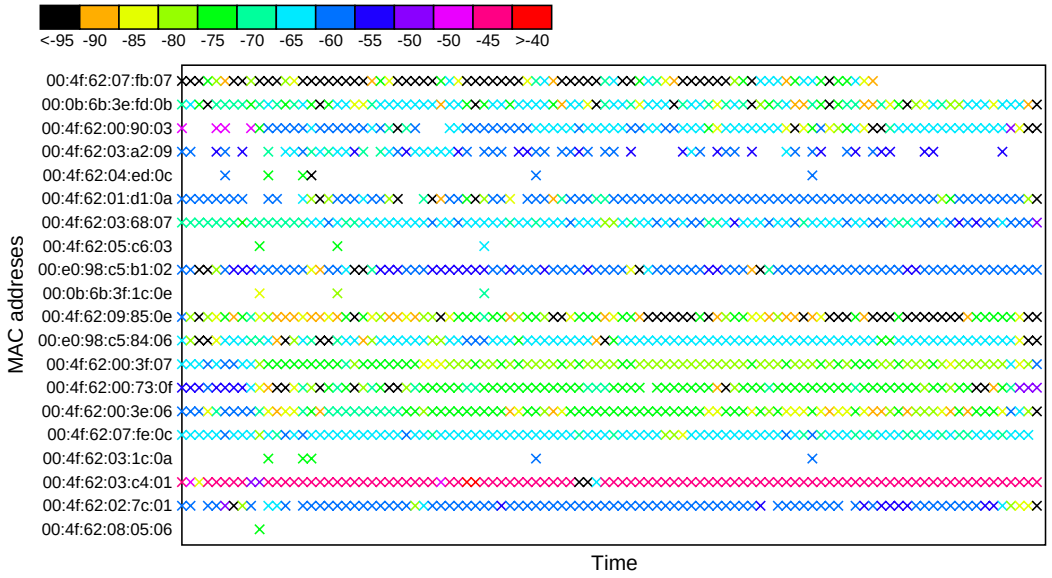
Druhá varianta vypouští barvy laděné do fialové barvy. Tímto vznikl celkem příjemně působivý graf, ale dle předpokladu šlo velice špatně rozpoznat žluté barvy.

Poslední varianta využívá barev celého spektra. Tímto se podařilo po vhodně zvoleném odstupu barev získat do palety grafu barvy, které jsou od sebe dobře rozlišitelné. Dále byla použita černá barva pro označení nejmenší síly signálu. Tímto lze jednoduše při prvním pohledu na graf vidět klienty, kteří mají signál na úrovni, kdy již v podstatě nemůžou komunikovat se sítí.

Rozsah hodnot, pro které byla vytvořena paleta, byl po konzultaci se správcem sítě zvolen od -95 do -40dBm. Vychází se z předpokladu, že -95dBm je nejnižší hodnota signálu, při které ještě může klient se sítí komunikovat. Naopak hodnoty signálu větší jak -40dBm nelze v praxi v podstatě docílit. Krok změny barvy v paletě byl stanoven na 5dBm. Změna signálu o tuto hodnotu může v praxi znamenat pokles nebo zvýšení rychlosti komunikace sítě s klientem.

Výsledný graf přístupů klientů do sítě, který je barvený dle absolutní hodnoty síly signálu klienta, je uveden na obrázku 4.6. Již při prvním pohledu na graf lze identifikovat prvního zobrazeného klienta, který má dlouhodobé problémy s připojením. Jeho síla signálu

se většinu času pohybuje na hranici, kdy je již klient odpojen od sítě. Může se také jednat o útočníka, který se snaží do sítě přihlásit z místa, kde nemá dostatečný signál.



Obrázek 4.6: Graf přístupů klientů barven dle absolutní hodnoty signálu (interval 1 měsíc)

Systém dále umožňuje kreslit stejný typ zmiňovaného grafu, který má pouze změněn způsob obarvení zobrazovaných křížků. V této variantě jsou křížky barveny dle odchylky síly signálu klienta od jeho průměru za zvolené časové období, pro které se kreslí graf. Tento typ grafu je uveden na obrázku 4.7.

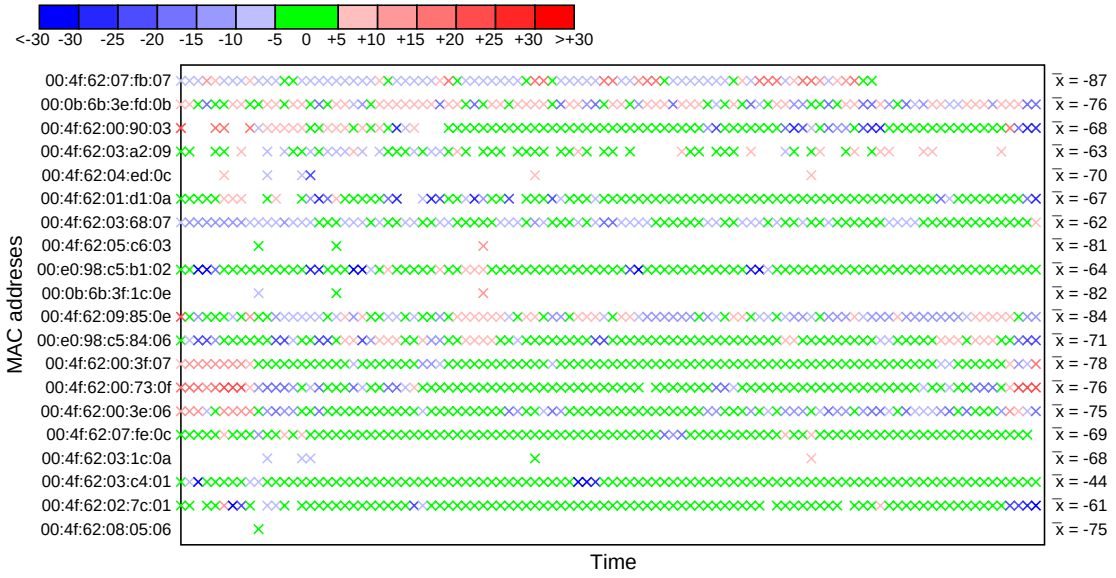
V tomto typu grafu je navíc na jeho pravé straně zobrazena průměrná hodnota síly signálu klienta. Pokud se síla signálu vykreslovaného křížku nepatrně liší od klientova průměru, je křížek kreslen zeleně nebo nádechem barvy do modré popř. červené. Signál, který je slabší než průměrný, postupně přechází do syté modré barvy. Naopak signál silnější než průměrný, přechází postupně do syté červené barvy.

Lze tedy při pohledu na graf vidět v sytých barvách klienty, kteří mohou být problémoví. Jejich signál se totiž neustále mění od jejich dlouhodobého průměru. Naopak klienti, kteří mají sílu signálu stále téměř stejnou, budou zobrazení zeleně popř. nevýraznou barvou s nádechem do modré popř. červené.

Pokud je nutné při kreslení grafu přístupů klientů agregovat hodnoty osy x, v případě, že jich je více než 100, pak se určení zda v daném intervalu bude křížek vykreslen používá prahování s prahem nastaveným na 50%.

4.4 Řešení přihlašování uživatele

V kapitole týkající se návrhu aplikace jsem zmínil, že pro řešení přihlašování v aplikaci reputačního systému použiji *session* proměnné a *cookies*. Session proměnné jsou použity



Obrázek 4.7: Graf přístupů klientů barven dle odchylky od průměru hodnoty signálu (interval 1 měsíc)

pro překonání bez stavovosti HTTP protokolu. Bez jejich použití by nebylo možné rozlišit, které požadavky pocházejí ze stejného klienta. Mechanismus session proměnných využívá cookies, ve kterých klient předává vygenerovaný řetězec serveru. Pomocí tohoto řetězce lze přesně určit relaci mezi klientem a serverem.

Vyžádání přístupových údajů od uživatele je provedeno pomocí formuláře. Vyžádaná data se posílají pomocí metody *post*. Metoda *post* má oproti metodě get tu výhodu, že se přihlašovací údaje neobjevují v URL[3] adresy stránky.

Zpracování přihlašovacích údajů provede skript *„login.php“*. Ověří zadaná data proti uloženým datům v databázi. Pokud přihlašovací údaje uživatele souhlasí, jsou naplněny session proměnné informacemi o uživateli. Pro zajištění větší bezpečnosti nejsou v databázi hesla uložena v čitelné podobě. Je uložen pouze jejich otisk vytvořený pomocí funkce *MD5*[4].

Reputační systém neumožňuje práci nepřihlášenému uživateli. Díky této vlastnosti je při načtení každé stránky uživatelského rozhraní testována přítomnost nastavených přihlašovacích session proměnných. Pokud proměnné neexistují, je uživatel přesměrován na stránku s přihlašovacím formulářem.

Vzhledem k tomu, že vstup neoprávněného uživatele do reputačního systému by mohl ohrozit chod celé reputované sítě, je nutné zajistit co největší bezpečnost aplikace. Pro tento případ bude v konečné verzi reputačního systému přenos stránek mezi serverem a klientem

[3] z anglického Uniform Resource Locator - jedná se o řetězec, který jednoznačně identifikuje jakýkoliv zdoj v síti Internet

[4] z anglického Message Digest 5 - jedná se o funkci, která vypočítá pro zadaná data otisk o pevné délce

zabezpečen pomocí HTTPS protokolu. Tímto se také vyřeší problém s posíláním hesla v otevřené podobě z klienta na server, které probíhá při odeslání přihlašovacího formuláře.

4.5 Inteligentní formuláře

Pojem inteligentní formuláře používám pro označení klasických XHTML formulářů, které jsou rozšířeny o několik funkcí.

Prvním rozšířením je kontrola vstupních polí formuláře již na straně klienta, nebo chceteli, webového prohlížeče. Lze aplikovat kontrolu na neprázdnost polí formuláře nebo různá porovnání zadaného vstupu s očekávaným formátem dat. Špatně vyplněný formulář se neposílá serveru, ale je vypsáno hlášení pomocí okna JavaScriptu. Tímto lze omezit zbytečné posílaní špatných dat serveru nebo předejít několikanásobnému poslaní prázdných formulářů. Pokud uživatel opraví data ve formuláři tak, že se shodují s očekávaným formátem, jsou data z formuláře odeslána serveru. Bohužel i v dnešní době nelze spoléhat na podporu JavaScriptu ve všech prohlížečích a je tedy nutné, aby skript, který formulář zpracuje, tyto data ještě jedou zkontroloval.

Dalším rozšířením formulářů je tzv. zapamatování zadaných hodnot. Veškerá data nelze kontrolovat na straně klienta. Jedná se především o tzv. integritní omezení, které jsou mezi daty zaneseny v databázovém schématu. Může se jednat např. o neshodující se cizí klíč nebo požadavek unikátnosti hodnoty. Neshodu s integritními omezeními lze zjistit až v okamžiku vložení dat do databáze, a tedy až po odeslání formuláře. Pokud se vložení hodnot nezdaří, je uživatel vyzván k opravě údajů a k jejich novému odeslání.

Klasické pojetí vstupních formulářů tento případ neřeší a uživateli je nabídnut nový prázdný formulář. Tato situace je velice nekomfortní v případě, že formulář měl více vstupních polí. Uživatel je tak nucen znovu vyplnit celý formulář i v případě, že mohl udělat chybu pouze v jednom ze vstupních polí. Inteligentní formuláře tento nedostatek řeší předvyplněním odeslaných dat a umožňují tak uživateli pouze opravit chybná pole.

Posledním rozšířením oproti klasické koncepci vstupních formulářů je přesměrování výstupu skriptu, který obsluhuje daný formulář. Tento skript negeneruje přímo výstup, ale pouze uloží (změní) požadovaná data v DB a přesměruje výstup na novou stránku. Na této stránce může být v případě chyby předvyplněný formulář, nebo v případě úspěchu operace hlášení o vykonané akci.

Cílem přesměrování je donucení internetového prohlížeče k zapomenutí odeslaných hodnot. Pokud se použije klasický přístup, tak v případě, že uživatel použije pro návrat k předchozí stránce s formulářem tlačítko *„Zpět“*, je tázán, zda si přeje odeslat zadaná data znovu. Bohužel dle vlastních zkušeností s vývojem internetových aplikací vím, že většina uživatelů tento dialog ani nečte a automaticky ho potvrzují. Tímto způsobem se do databáze mohou dostávat duplicitní data, která uživatel několikrát poslal, aniž by tak chtěl učinit. Po objevení zmiňovaného potvrzovacího dialogu většina neznalých uživatelů ztrácí

orientaci v internetové aplikaci. Postup s přesměrováním tento dialog plně vyřadí a je tedy možné jednoduše procházet historii stránek bez znovuodesílaní dat.

Velkou inspirací pro koncept inteligentních formulářů pro mě byla kniha [23], kterou napsal pan Kosek.

4.6 Jazykové mutace

Při implementaci jazykových mutací uživatelského rozhraní jsem dbal především na připravení systému pro jednoduchý překlad do dalších jazyků. Obecné řešení jazykových mutací programů vždy vede na použití určitého číselníku, který pod unikátním identifikátorem obsahuje větu nebo slovo ve zvoleném jazyku.

V mojí implementaci jsem zvolil umístění vždy jedné jazykové mutace do jednoho souboru typu XML. Pokud bude nutné vytvořit překlad uživatelského rozhraní do dalšího jazyka, je třeba pouze přeložit tento soubor. V následujícím textu uvádím část souboru *„CZ-lang.xml“*, který obsahuje český překlad uživatelského rozhraní:

```
<pages>
  <page name=„login.php\ lang=„CZ\>
    <entry id=„login\>Jméno</entry>
    <entry id=„password\>Heslo</entry>
    <entry id=„loginsubmit\>Přihlásit</entry>
    <entry id=„er1\>Bohužel zadaný uživatel ...</entry>
    <entry id=„er2\>Litujeme, ale ...</entry>
    <entry id=„er3\>Litujeme, ale nepodařilo se ...</entry>
  </page>
  .
  .
</pages>
```

Uvedená ukázka kódu zobrazuje překlad stránky, kterou sestavuje skript *„login.php“*. Značky `<entry>` určují jednotlivé položky k překladu. Každá značka `<entry>` je reprezentována jednoznačným identifikátorem s rozsahem platnosti pro překládanou stránku. Systém jazykových mutací umožňuje překládat i hlášení, která se na obrazovku vypisují pomocí jazyka JavaScript.

Texty v aktuální jazykové mutaci následně používají XSL šablony. Na následující ukázce lze vidět import šablon ze souboru *„templates.xsl“*, které zajistí načtení zvolené jazykové mutace dle vybraného jazyka. Dále je uvedena ukázka výpisu přeloženého textu pro identifikátor login.

```
<xsl:import href=„templates.xsl\/>
<xsl:value-of select=„$lf[@id='login']\/>
```

4.7 Zajištění senzorových dat

Jak jsem již v úvodu diplomové práce uvedl, jednou ze součástí reputačního systému je jeho senzorová část. Tato část zajišťuje sběr informací o sledovaných entitách. V našem případě se tedy jedná o sběr informací z jednotlivých přístupových bodů sítě repuNET.

Při koncipování rozsahu práce bylo přijato omezení hardwaru sítě pouze na přístupové body od firmy MikroTik. Při implementaci senzorových skriptů je tedy nutné vybrat vhodný způsob, jak data z jednotlivých přístupových bodů číst a přenášet na server. Při analýze možností, které lze využít pro čtení dat z přístupových bodů, byly nalezeny dvě možnosti. Data lze získat využitím protokolu SNMP[5] nebo využitím protokolu SSH[6].

Bohužel drtivá většina přístupových bodů firmy MikroTik v současné době podporuje pouze protokol SNMP verze 1. Tato verze je v dnešní době označovaná již jako zastaralá. Hlavním problémem této verze je slabé zabezpečení. Dochází např. k přenášení hesel v nezašifrované podobě. Další nedostatek při použití SNMP je, že přístupové body firmy MikroTik umožňují pomocí SNMP pouze čtení. Jakékoliv změny v nastavení přístupových bodů by pak bylo nutné řešit jinou cestou.

Pro implementaci komunikace byl tedy zvolen protokol SSH. Využitím tohoto protokolu je možné data z přístupových bodů číst i do přístupových bodů zapisovat. Další výhodou je, že SSH protokol zajistí bezpečnou komunikaci mezi dvěma počítači.

Aby sběr informací z přístupových bodů mohl probíhat zcela autonomně, je vhodné využít autentizace pomocí veřejných klíčů. Využitím této metody bude umožněno bezpečné přihlášení na vzdálený přístupový bod, aniž by bylo nutné zadávat heslo.

Nastavení komunikace mezi routerboardem s operačním systémem RouterOS a serverem, který bude daná data shromažďovat uvedu v několika následujících bodech.

1. Vygenerování klíčů pro SSH. Vygeneruje se dvojice klíčů (veřejný a privátní). Privátní klíč se ponechá na centrálním serveru, který získává data z přístupových bodů.

2. Veřejný klíč se nakopíruje pomocí FTP na všechny spravované přístupové body v síti repuNET. K tomuto účelu slouží skript *„ftp.py“*, který z databáze získá seznam přístupových bodů, vyžádá si zadání hesel na přístupové body a veřejný klíč na přístupové body nahraje.

3. Po předchozím kroku již můžeme z centrálního serveru přistupovat na jednotlivé přístupové body bez nutnosti zadávat heslo. Komunikace pro autonomní sběr dat z přístupových bodů je tedy nastavena.

[5] z anglického Simple Network Management Protocol - jedná se o protokol, který má ulehčit správu větších sítí

[6] z anglického Secure Shell - klient/server protokol nad TCP/IP, který umožňuje bezpečnou komunikaci mezi dvěma počítači

Sběr dat z přístupových bodů zajišťuje skript *„readstat.py“*. Tento skript je volán v pravidelných intervalech pomocí démona CRON[7]. Skript získá relevantní data z přístupových bodů a tyto data uloží pro další zpracování do databáze na centrálním serveru.

[7]CRON je systémový nástroj pro spouštění různých programů v předem definovaném čase a intervalech.

Kapitola 5

Testování

Veškerý vývoj reputačního systému probíhal na vyhrazeném serveru v laboratoři BUSlab. Využitím opravdového serveru má výhodu především v získání zkušeností s reálným nasazením reputačního systému.

Zde bych chtěl poděkovat Danielu Cvrčkovi, který průběžně testuje mojí implementaci a svými připomínkami přispívá ke zlepšování celého systému. Pro mapování vývoje reputačního systému byla založena *DokuWiki*, ve které bylo možné naleznout seznam nalezených chyb a seznam rozšíření, které jsou požadované zadavatelem projektu. Po havárii serveru se bohužel nepodařilo data obnovit.

Uživatelské rozhraní a jeho zobrazení je v současné době testováno na prohlížečích *Internet Explorer*, *FireFox*, *Mozilla* a *Opera*. Při testování bylo nalezeno několik chyb v zobrazení především na prohlížečích, které běží na platformě Linux.

Poslední verze systému uživatelského rozhraní byla testována i na platformě Mac Os. Zde se u různých grafů objevují problémy s jejich vykreslením. Graf se vykreslí, ale vykreslení trvá i několik desítek vteřin. Problému se budeme dále věnovat, ale s největší pravděpodobností se jedná o nedokonalou implementaci zobrazovacího modulu internetového prohlížeče.

Kapitola 6

Nasazení

Vzhledem k tomu, že v době odevzdání diplomové práce stále není dokončené jádro reputačního systému, neobsahuje ani uživatelské rozhraní veškeré prvky, které budou použity v „ostré“ verzi reputačního systému. Po dokončení systému bude pro systém společně s uživatelským rozhraním vytvořen instalátor. V současné době se také zvažuje distribuce systému pomocí „image“ pro virtuální stroj VMware[1] více na adrese.[2] Využitím tohoto způsobu bude možné vytvořený produkt jednoduše prezentovat a nechat potencionálnímu zákazníkovi otestovat, aniž by byl nucen celý systém instalovat a konfigurovat.

Z tohoto důvodu uvedu pouze požadavky nově implementovaného uživatelského rozhraní, které je nutné splnit pro jeho spuštění. Následující popis poskytuje návod, jak lze systém spustit na platformě Microsoft Windows. Veškeré potřebné programy pro běh uživatelského rozhraní existují i ve vydáních pro jiné platformy. Pravděpodobně se bude lišit pouze způsob konfigurace jednotlivých programů.

Nejnutnější prekvizitou pro běh programu je server Apache verze alespoň 1.3.x. V konfiguraci webového serveru je třeba oproti standardní instalaci povolit moduly pro PHP, rewrite a gzip. Dále je třeba povolit pro uživatelské rozhraní používání souboru „.htaccess“. Toho lze dosáhnout vložením následujících řádků do souboru „httpd.conf“:

```
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule php5_module CESTA DLE INSTALACE PHP/php5apache.dll
LoadModule gzip_module modules/ApacheModuleGzip.dll

AddModule mod_rewrite.c
AddModule mod_php5.c
AddModule mod_gzip.c

<Directory „CESTA DLE INSTALACE APACHE/htdocs/repunet\>
    Options Indexes Includes FollowSymLinks MultiViews
```

[1] jedná se o softwarový produkt, který virtualizuje hardware počítače

[2] `www.vmware.com`

```
    AllowOverride All
    Order allow,deny
    Allow from all
</Directory>
```

Dále je nutné nainstalovat interpret jazyka PHP. Požadovaná verze je minimálně 5.1.2. V konfiguraci PHP, která je uložena v souboru „php.ini“, je třeba povolit rozšíření pro XSL a PDO s podporou databáze PostgreSQL. Tohoto lze docílit vložením následujících řádků do souboru „php.ini“.

```
extension=php_xsl.dll
extension=php_pdo.dll
extension=php_pdo_pgsql.dll
```

Poslední prekvizitou pro běh uživatelského rozhraní je databázový systém PostgreSQL. Pro současnou implementaci uživatelského rozhraní byla použita verze 8.1. Na přiloženém médiu lze v adresáři *DB* naleznout skript pro vytvoření schématu databáze a skript s demo daty projektu. Tyto skripty lze využít pro otestování uživatelského rozhraní.

Po instalaci databáze je třeba nastavit přístupové údaje databáze do souboru „header.php“, který je součástí implementace uživatelského rozhraní. Na přiloženém médiu je umístěn v adresáři *GUI/repunet/inc*.

Kapitola 7

Závěr

Cílem mé diplomové práce bylo především analyzovat současný reputační systém a možnosti rozšíření jeho uživatelského rozhraní.

Ve své práci jsem se snažil zachytit všechny etapy vývoje aplikace a zdokumentovat důležitá rozhodnutí, ke kterým jsem při řešení projektu přistoupil. Při řešení projektu jsem se vždy snažil vycházet z trendů, které jsou v dané oblasti používány a nasazovány. Po jejich osvojení jsem přistoupil k jejich nasazení pro řešení aktuálního problému.

Vznikl tak úplně nový systém uživatelského rozhraní pro reputační systém. Toto rozhraní je díky využití šablon velice jednoduše rozšiřitelné a budoucí vývoj rozhraní by měl již směřovat především cestou úprav těchto šablon.

Vytvořený systém také umožňuje autentizaci a autorizaci uživatelů systému. Řízení práv uživatelů je implementováno jednoduchým způsobem, který ale pokrývá nároky kladené na tuto část systému zadavatelem projektu.

Díky vhodně zvolené technologii pro kreslení grafů a topologie sítě bude v budoucnu možné grafy a mapu rozšířit o další interaktivitu. Je možné např. uvažovat rozkliky klientů z grafu pro zobrazení dalších podrobnějších dat o klientech nebo využit animací.

Celému uživatelskému rozhraní bylo po důkladné analýze a návrhu vtisknuto dělení zobrazovaných informací na informace týkající se bezpečnosti a informace týkající se nastavení sítě. Tento krok velice přispěl k čitelnosti a přehlednosti uživatelského rozhraní.

Nelze také opomenout na systém nápovědy, který se snaží uživatele provést ovládáním celého systému. Důležitým prvkem je také možnost využít více jazykových mutací. Tato vlastnost by v budoucnu měla projektu ulehčit vstup na zahraniční trh.

Při řešení projektu jsem získal mnoho zkušeností s týmovým vývojem aplikací a seznámil jsem se také s několika nástroji pro podporu týmové spolupráce.

7.1 Chyby a problémy při řešení projektu

Při řešení projektu jsem narazil na několik zásadních pochybení, které bych chtěl na tomto místě ve stručnosti uvést.

Dle mého názoru bylo zvoleno nevhodné personální obsazení týmu. Do projektu byl jako druhý externí programátor přizván původní autor systému Petr Blahák, který je v současné době zaměstnán a již nestuduje. Během své několikaleté praxe jsem pracoval v několika týmech zabývajících se tvorbou softwaru. Až do této doby jsem se nesetkal s podobným typem člověka. Nebylo možné dotyčného donutit dodržet jakýkoliv termín při práci na projektu. Bohužel i tento fakt přispěl k tomu, že v době odevzdání diplomové práce není stále hotové funkční jádro reputačního systému a zpětné vazby. Pokud bych mohl volit, snažil bych se do projektu zapojit lidské zdroje, které mají dostatek času a chuti věnovat se projektu.

Další zkušeností byla havárie počítače, na kterém probíhal vývoj celého systému. Díky prozřetelnosti a opatrnosti se podařilo z pravidelných záloh obnovit více jak 80% dat.

Dle mého názoru v projektu nejvíce chybělo projektové řízení. Snažil jsem se při implementaci a dokumentování tvořit vlastní plán s časovými milníky. Bohužel snažit se projekt plánovat a zároveň v něm působit v roli analytika, návrháře, programátora a testera je nad možnosti obyčejného člověka. Nemohu ani opomenout, že v takovéto pozici jsem neměl žádný nástroj, jak případně motivovat spoluprogramátora k práci. V praxi je toto téměř vždy řešeno finančním ohodnocením, které v tomto projektu nepřicházelo v úvahu.

Domnívám se, že pro řešení projektu by bylo vhodné přizvání dalšího člověka. Ten by měl za úkol projektové plánování a řízení. Zároveň by působil jako nestranný arbitr a dbal na dodržování termínů od každého člena týmu. Projektové řízení a případné řízení rizik by dokázalo rychleji odhalit nedostatky projektu a bylo by možné na ně lépe reagovat.

I přes zmiňované nedostatky se podařilo projekt dovést do pokročilé fáze. Plánuje se pokračovaní v projektu a dovedení aplikace do komerční podoby. Při malém průzkumu trhu byl zjištěn potencionální zájem o námi vyvíjený systém.

7.2 Další možná rozšíření projektu

Současný projekt nabízí mnohá rozšíření. Velice vhodné by bylo upravit reputační systém tak, aby nebyl závislý na použitém hardwaru sítě. Takto upravený systém by bylo možné nasadit i v jiných bezdrátových sítích než jsou *WiFi* sítě.

Hlavním a nutným rozšířením současné implementace projektu je agregace senzorových dat. Současný systém pracuje s daty od posledního vzorku maximálně měsíc zpět v čase. Starší data je nutné agregovat a zmenšit tak jejich velikost. S tímto souvisí i další implementace funkcí pro práci s agregovaným typem dat.

Nutností je také dokončení jádra reputačního systému a systému zpětné vazby, pro které bude třeba dle nových požadavků upravit stávající uživatelské rozhraní.

Mezi další zvažovaná rozšíření patří zobrazení měřítka mapy a umožnění vložení mapového podkladu skutečné oblasti s přístupovými body.

Dále se plánují již zmiňované rozšíření interaktivity grafů týkající se přechodů mezi grafy, zvětšování a zmenšování počtu hodnot grafu a případné prostupy do agregovaných dat.

Myslím si, že by bylo vhodné také propojit systém s databází registrací klientů. Tímto by bylo možné např. místo výpisu MAC adres zobrazovat přímo jméno klienta a jeho adresu. Dále by pak bylo možné v systému ovlivňovat nastavení služeb např. dle platební historie klienta.

Literatura

[1] Hargaš, P.: *Reputačné systémy.* Vysoké učení technické v Brně, Fakulta informačních technologií, 2006. 67 s. Vedoucí diplomové práce doc. Ing. Daniel Cvrček, Ph.D.

[2] Blahák, P.: *Reputační systémy ve WiFi sítích.* Vysoké učení technické v Brně, Fakulta informačních technologií, 2006. 59 s. Vedoucí diplomové práce doc. Ing. Daniel Cvrček, Ph.D.

[3] AirDefense, Inc.: *AirDefense [online].* 2001-2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.airdefense.net/`.

[4] Lockhart, A.: *Snort-wireless [online].* 2003-2005 [cit. 2006-12-29]. Dostupný z WWW: `http://www.snort-wireless.org/`.

[5] Osborne, M.: *WIDZV 1.5 – The Wireless IDS for 802.11b [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.loud-fat-bloke.co.uk/tools/widzv1.5.zip`.

[6] MikroTik: *The Dude [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.mikrotik.com/thedude.php`.

[7] MikroTik: *RouterOS [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.mikrotik.com/software.html`.

[8] Bos, B.: *What is a good standard? [online].* 2003 [cit. 2007-05-11]. Dostupný z WWW: `http://www.w3.org/People/Bos/DesignGuide/introduction.html`.

[9] Wikipedia: *World Geodetic System [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://en.wikipedia.org/wiki/WGS84`.

[10] Converse, T. – Park, J. – Morgan, C.: *PHP 5 and MySQL Bible.* Wiley Publising, Inc., 2004. 1083 s. ISBN 0-7645-5746-7.

[11] Glass, M. – Le, Y. – Naramore, E. – Mailer, G. – Stolz, J. – Gerner, J.: *Beginning PHP, Apache, MySQL Web Development.* Wiley Publising, Inc., 2004. 723 s. ISBN 0-7645-5744-0.

[12] MySQL AB: *MySQL 5.0 Reference Manual [online].* 1997-2006 [cit. 2006-12-29]. Dostupný z WWW: `http://dev.mysql.com/doc/refman/5.0/en/`.

[13] IBM Corporation: *Informační centrum produktu IBM DB2 [online].* 1993-2006 [cit. 2006-12-29]. Dostupný z WWW: `http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp`.

[14] Oracle: *Oracle Database Documentary Library [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.oracle.com/pls/xe102/homepage`.

[15] PostgreSQL Development Group: *PostgreSQL 8.0.9 Documentation [online].* 1996-2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.postgresql.org/docs/8.0/interactive/index.html`.

[16] Harold, E.: *XML Bible Second Edition.* Hungry Minds, Inc., 2001. 1249 s. ISBN 0-7645-4760-7.

[17] Pfaffenberger, B. – Schafer, S. – White, C. – Karow, B.: *HTML, XHTML, and CSS Bible 3rd Edition.* Wiley Publising, Inc., 2004. 843 s. ISBN 0-7645-7718-2.

[18] Staníček, P.: *CSS Kaskádové styly Kompletní průvodce.* Computer Press, 2003. 178 s. ISBN 80-7226-872-4.

[19] Prokop, M.: *Magie barev na webu [online].* 2001 [cit. 2007-05-12]. Dostupný z WWW: `http://interval.cz/clanky/magie-barev-na-webu-zaklady-teorie/`.

[20] SVG Working Group: *Scalable Vector Graphics (SVG) [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://www.w3.org/Graphics/SVG/`.

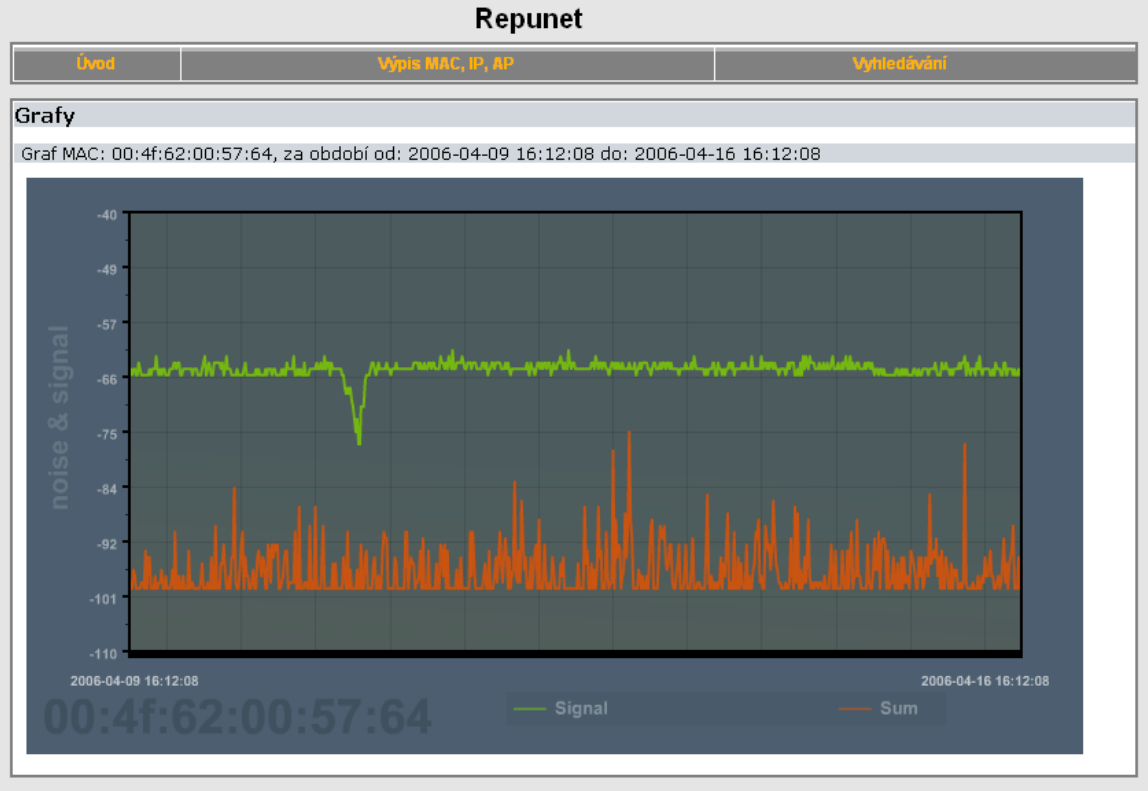
[21] Hirš, P.: *Systém pro organizaci webových záložek.* Vysoké učení technické v Brně, Fakulta informačních technologií, 2005. 35 s. Vedoucí diplomové práce doc. Ing. Vladimír Čech

[22] SVG Wiki: *SVG-Wiki [online].* 2006 [cit. 2006-12-29]. Dostupný z WWW: `http://wiki.svg.org/Main_Page`.

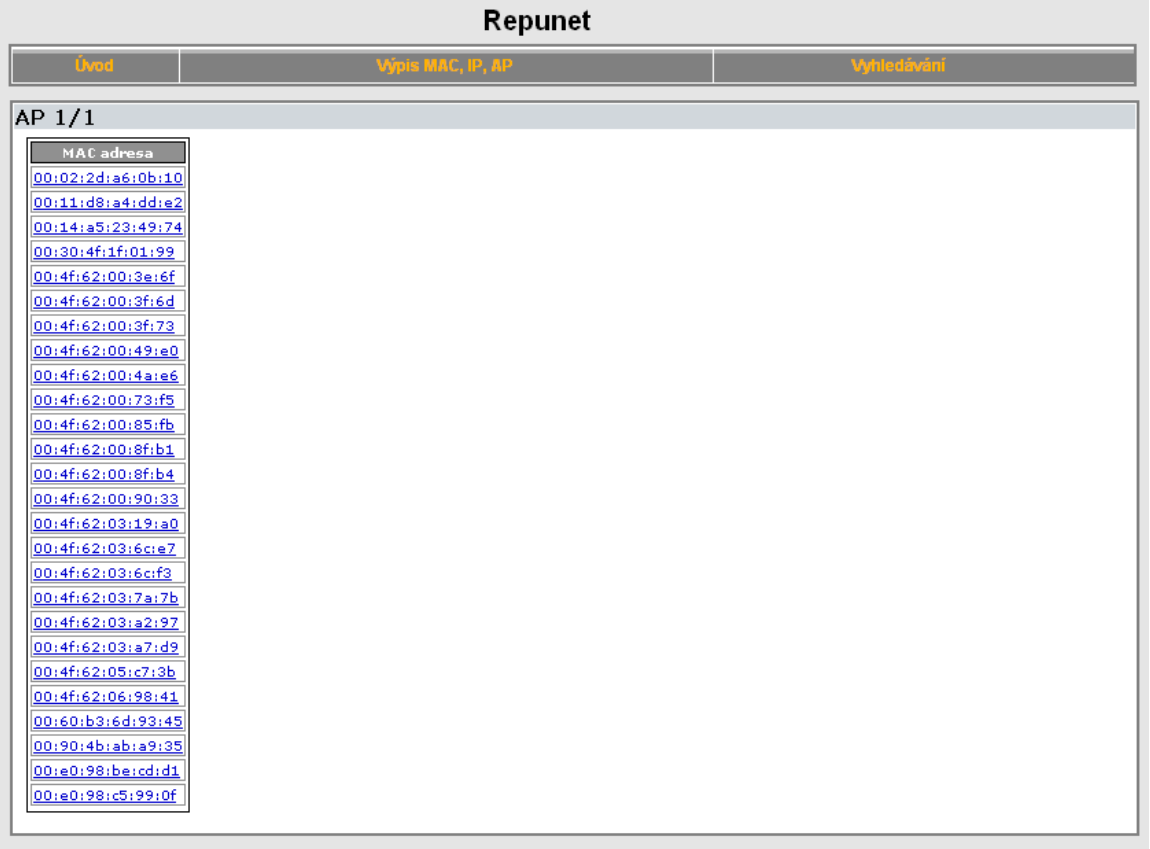
[23] Kosek, J.: *PHP - tvorba interaktivních internetových aplikací.* Grada Publishing, 1998. 490 s. ISBN 80-7169-373-1.

Dodatek A

Ukázka původního uživatelského rozhraní



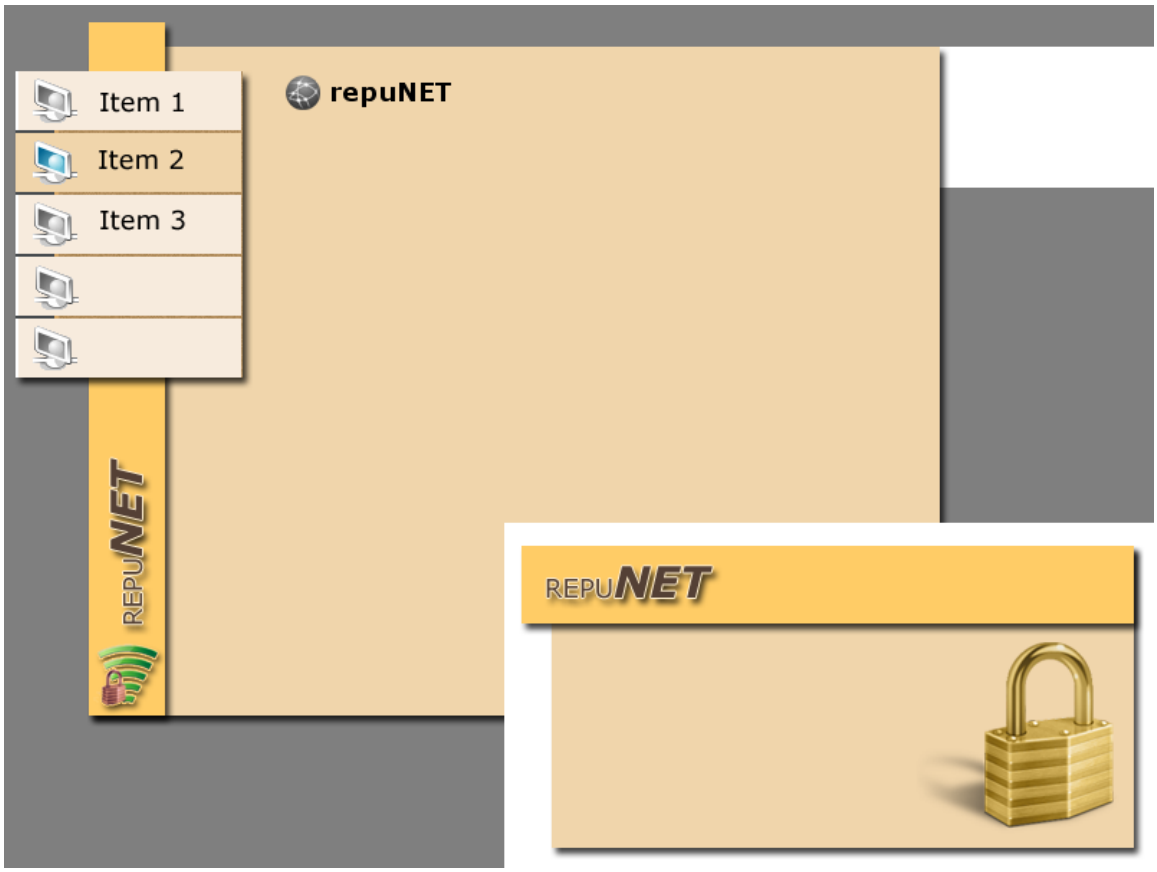
Obrázek A.1: Původní vzhled aplikace při vykreslení grafu síly signálu



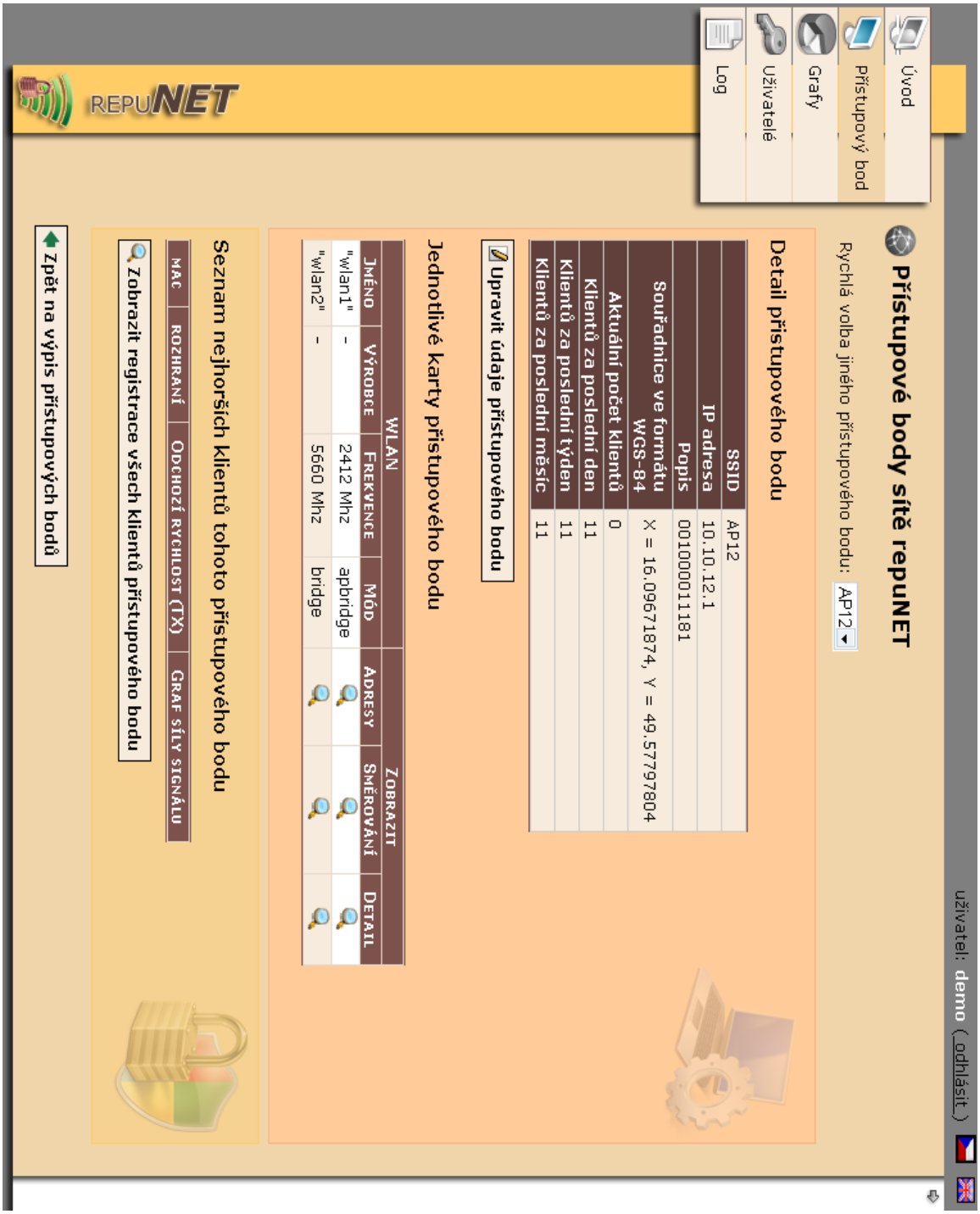
Obrázek A.2: Původní vzhled aplikace při výpisu klientů přistupového bodu

Dodatek B

Prvotní návrh vzhledu uživatelského rozhraní



Obrázek B.1: Návrh vzhledu aplikace vytvořený v programu Photoshop



Obrázek B.2: Zobrazení detailu přístupového bodu poslední verze uživatelského rozhraní